

Memory Diagram Walkthrough: I

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

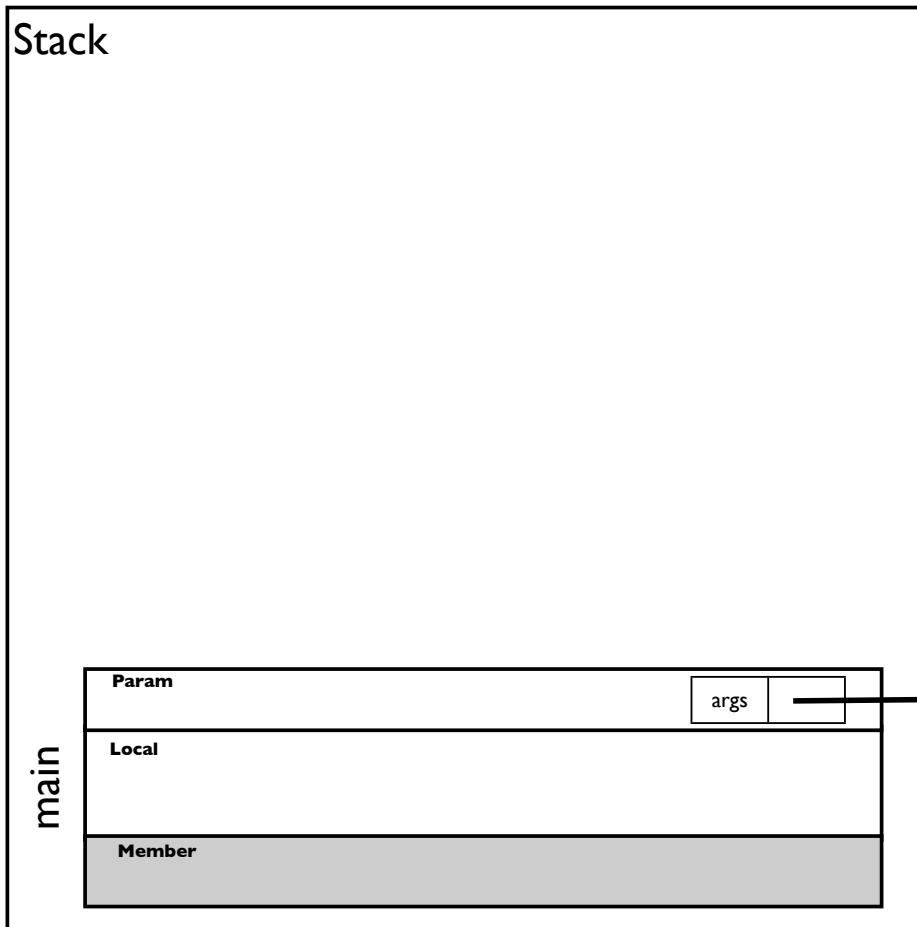
    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

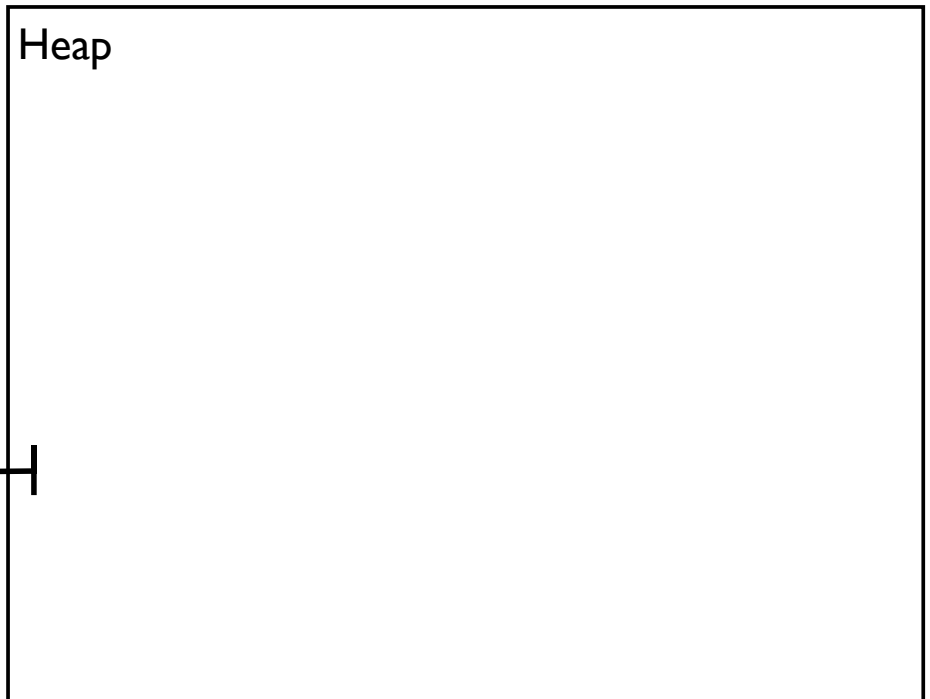
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack



Heap



Memory Diagram Walkthrough: 2

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

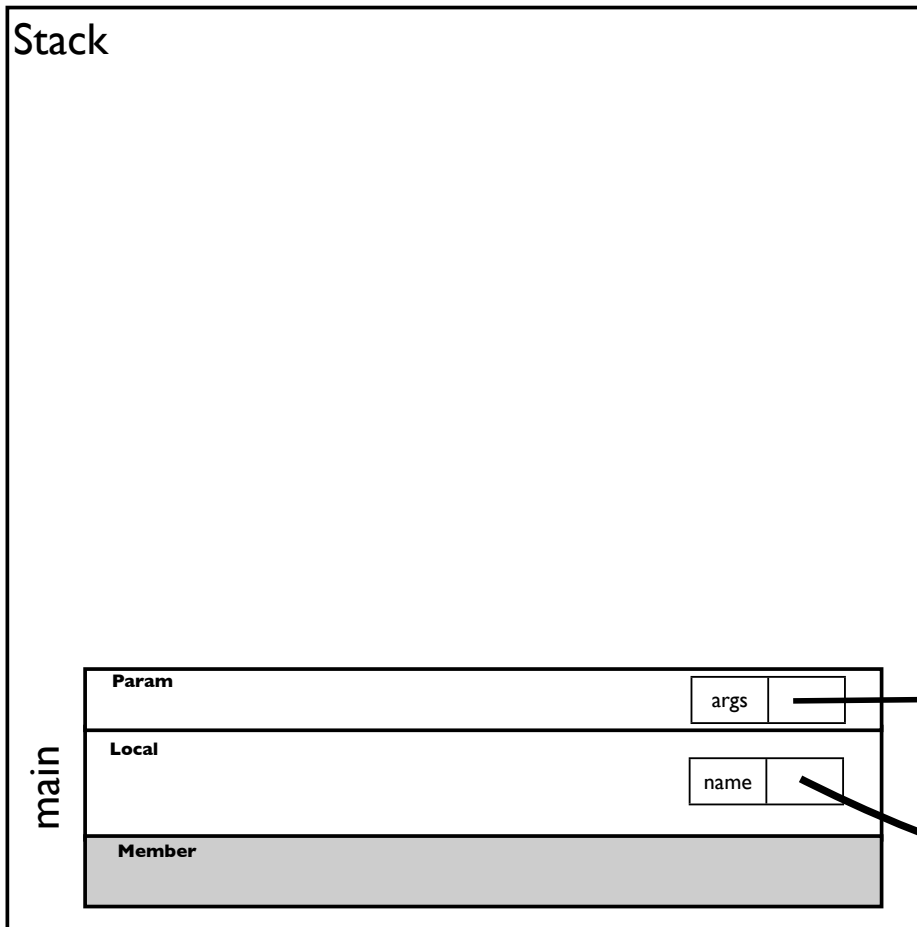
    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

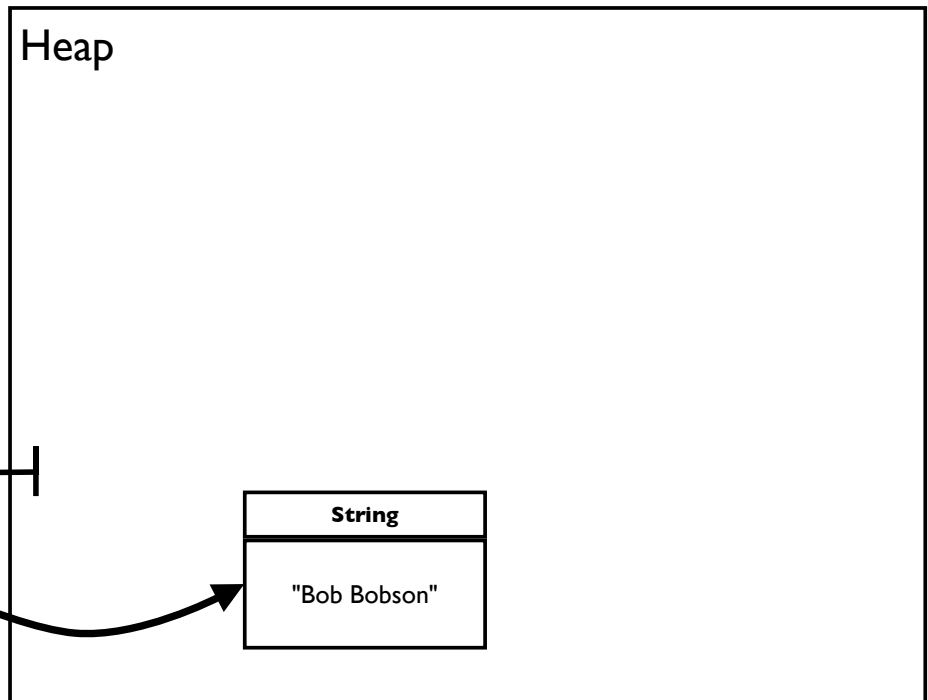
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack



Heap



Memory Diagram Walkthrough: 3

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

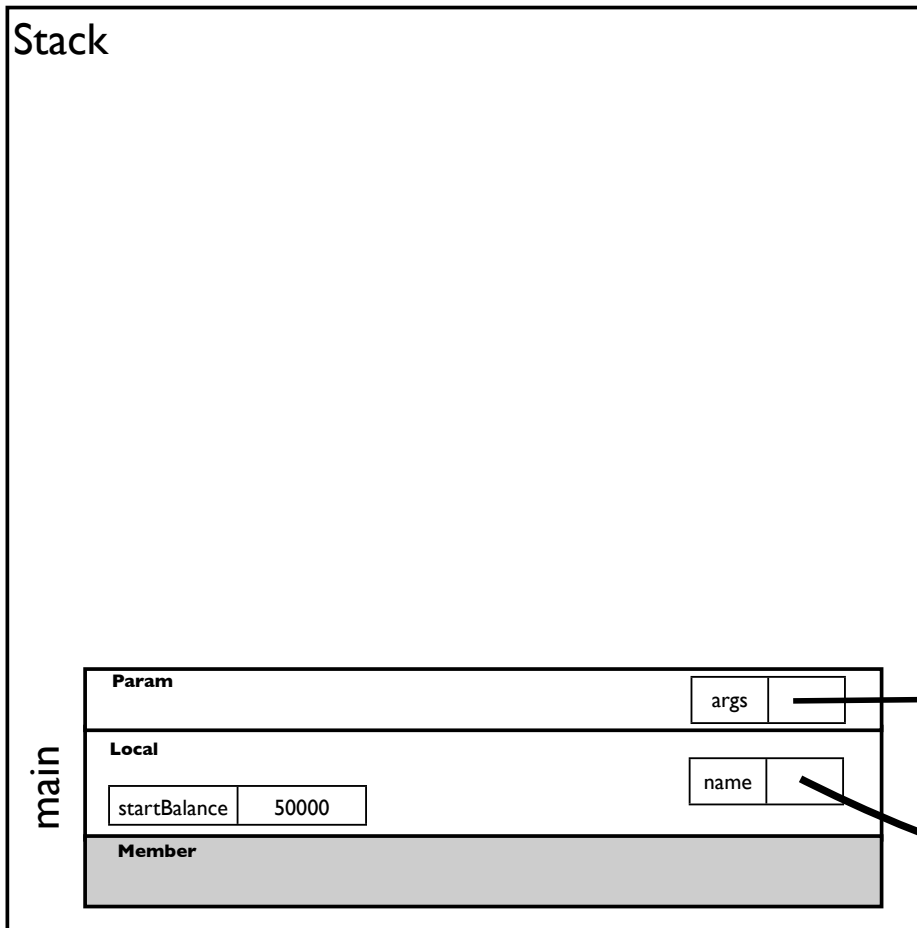
    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

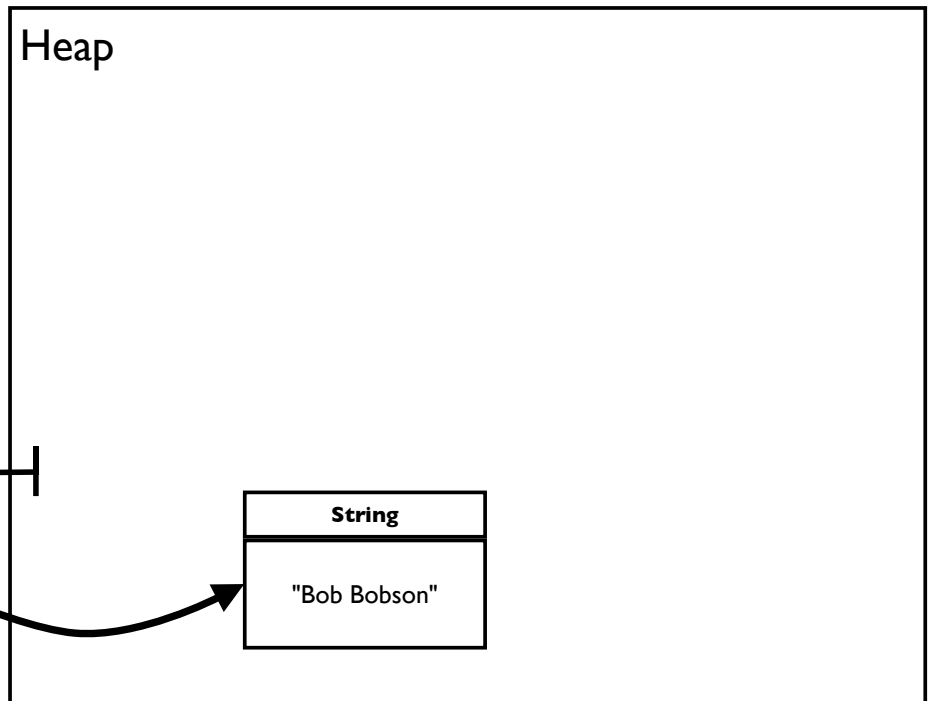
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack



Heap



Memory Diagram Walkthrough: 4

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

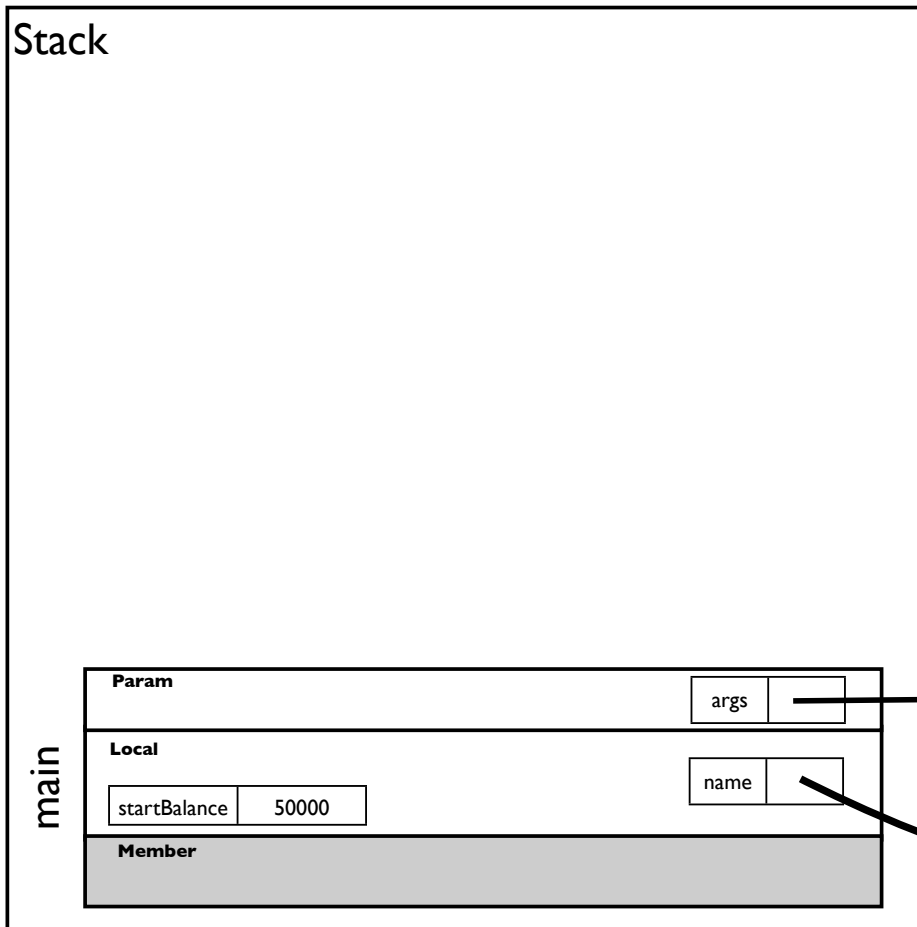
    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

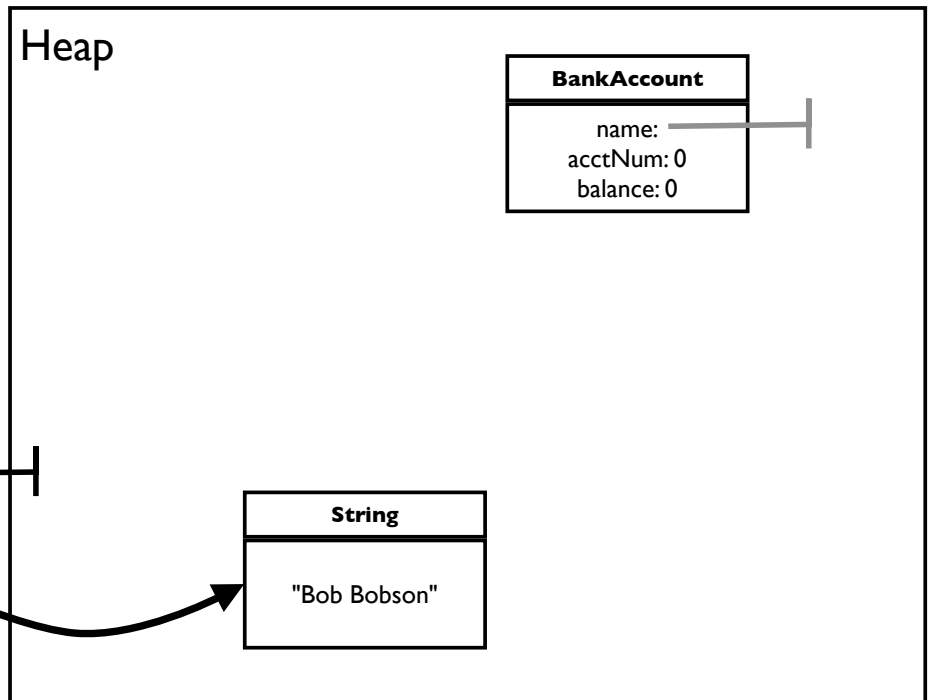
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack



Heap



Memory Diagram Walkthrough: 5

BankApplication.java

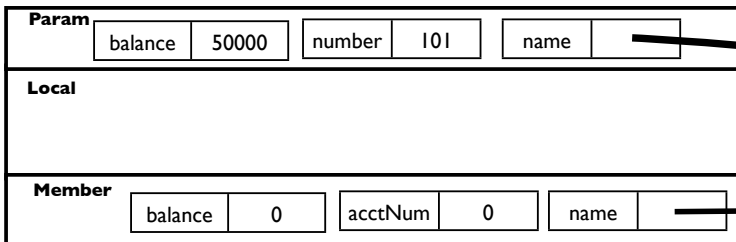
```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }
    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }
    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

Stack

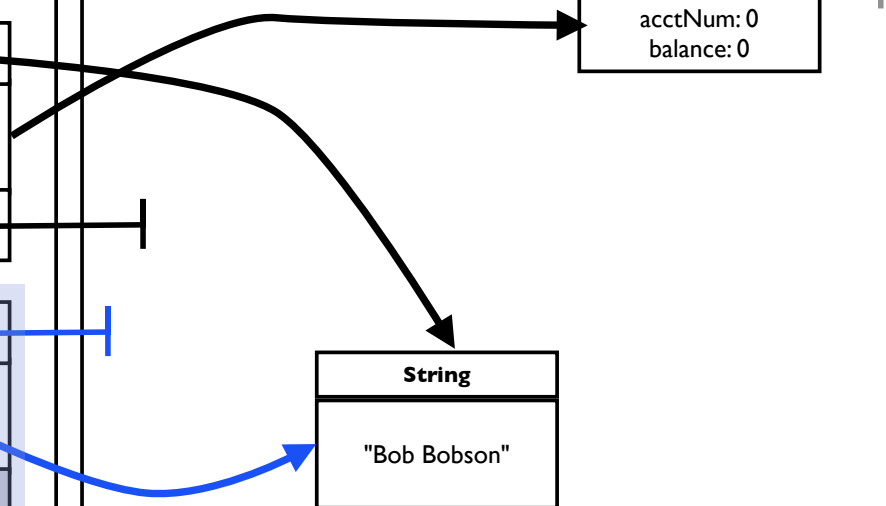
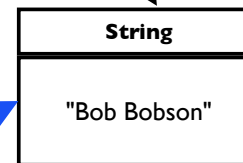
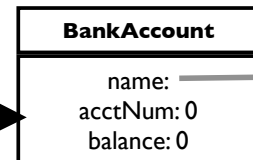
BankAccount(
String,int,int)



main: 3



Heap



Memory Diagram Walkthrough: 6

BankApplication.java

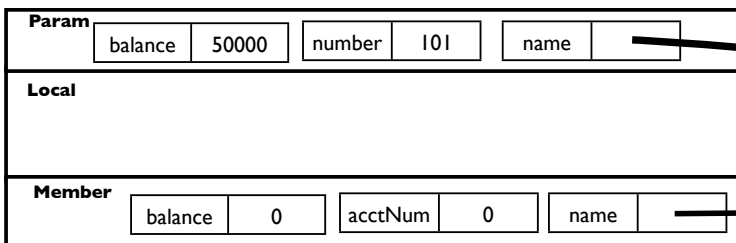
```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }
    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }
    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

Stack

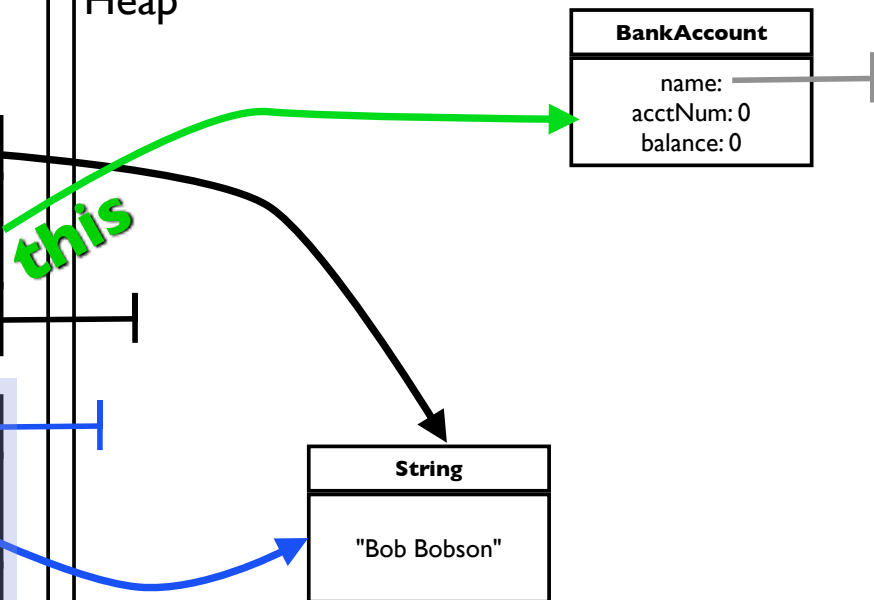
BankAccount(
String,int,int)



main: 3



Heap



Memory Diagram Walkthrough: 7

BankApplication.java

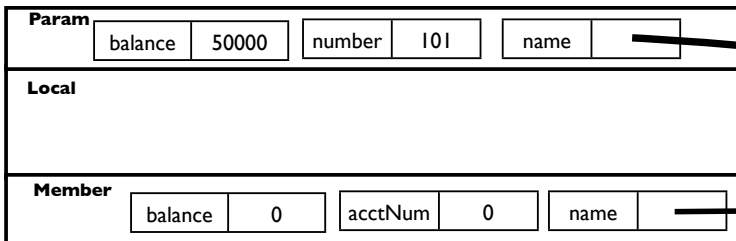
```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }
    public BankAccount(String name, int number, int balance) {
        name = name;        acctNum = number;        balance = balance;
    }
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }
    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack

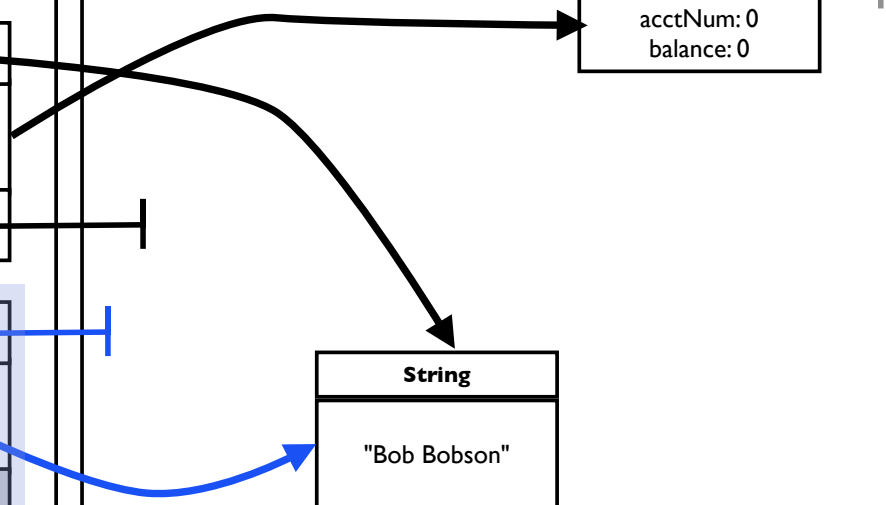
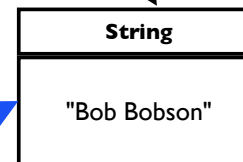
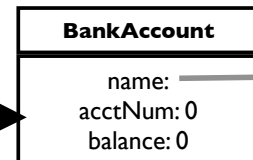
BankAccount(
String,int,int)



main: 3



Heap



Memory Diagram Walkthrough: 8

BankApplication.java

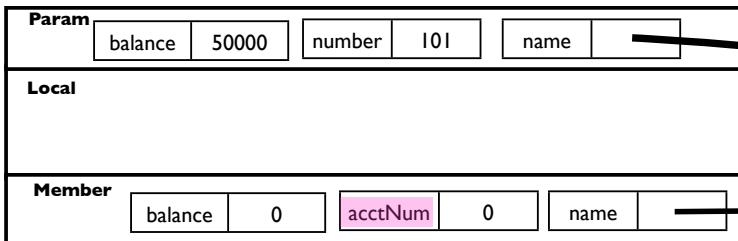
```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }
    public BankAccount(String name, int number, int balance) {
        name = name; acctNum = number; balance = balance;
    }
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }
    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

Stack

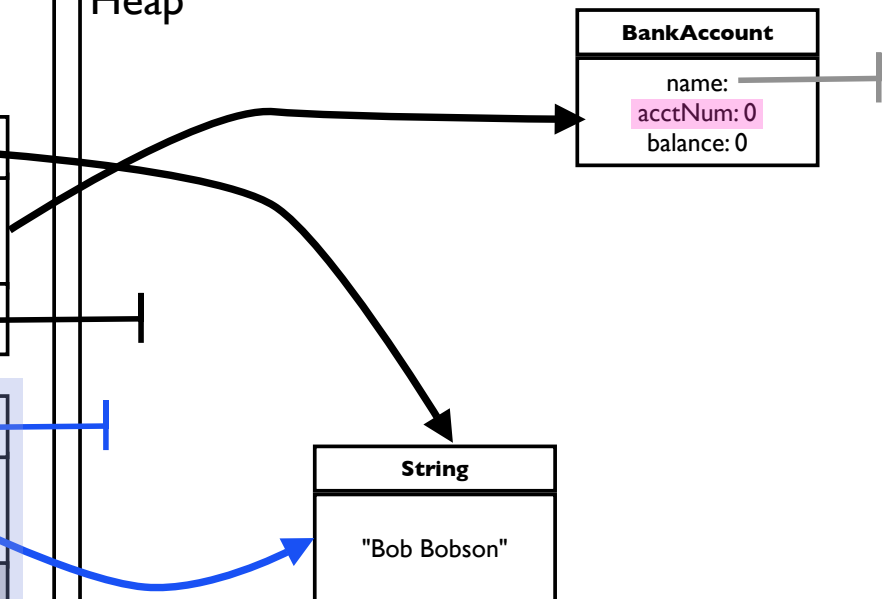
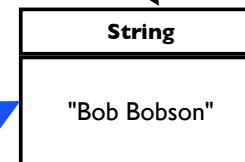
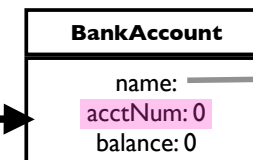
BankAccount(
String,int,int)



main: 3



Heap



Memory Diagram Walkthrough: 9

BankApplication.java

```
public class BankApplication {  
    public static void main(String[] args) {  
        String name = "Bob Bobson";  
        int startBalance = 50000;  
        BankAccount ba1 = new BankAccount(name, 101, startBalance);  
        int newBalance = ba1.makeDeposit(20000);  
        BankAccount ba2 = new BankAccount(name, 102);  
        ba1.transferMoney(ba2, 15000);  
    }  
}
```

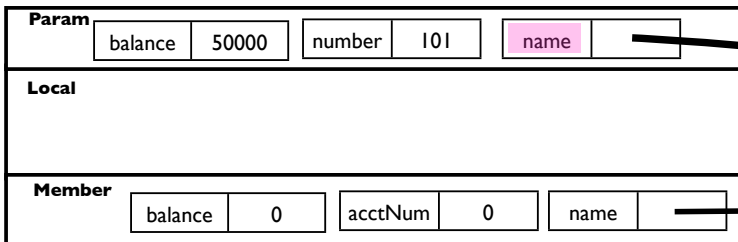
BankAccount.java

Andrew Weinrich CS302 weinrich at wisc.edu BankAccountWalkthrough.key

```
public class BankAccount {  
    public BankAccount(String name, int number) {  
        this.name = name;    this.acctNum = number;    this.balance = 0;  
    }  
    public BankAccount(String name, int number, int balance) {  
        name = name;        acctNum = number;        balance = balance;  
    }  
    public int makeDeposit(int amount) {  
        int newBalance = this.getBalance() + amount;  
        this.setBalance(newBalance);  
        return newBalance;  
    }  
    public void transferMoney(BankAccount b, int amount) {  
        this.setBalance(this.getBalance() - amount);  
        b.setBalance(b.getBalance() + amount);  
    }  
    private void setBalance(int newAmount) {    balance = newAmount;    }  
    public int getBalance() {    return balance;    }  
}
```

Stack

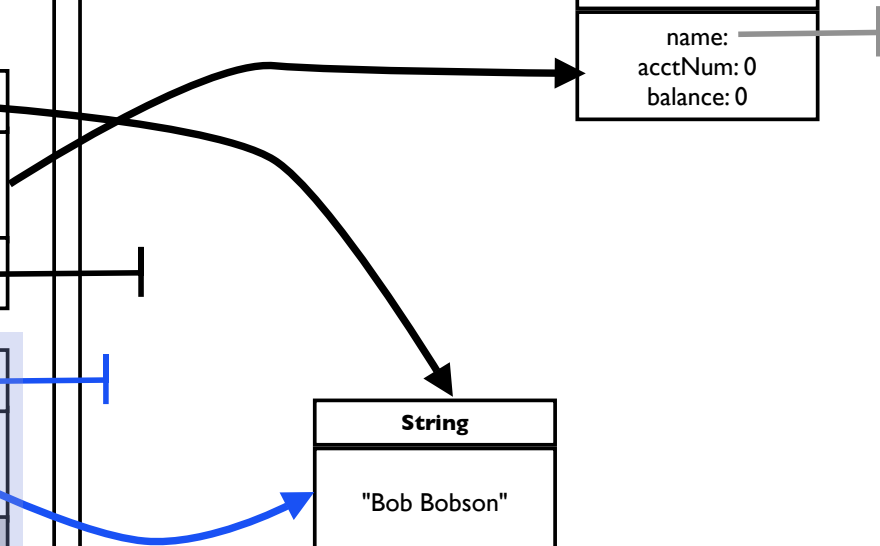
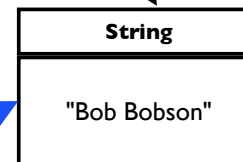
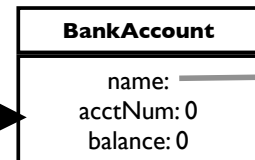
BankAccount(
String,int,int)



main: 3



Heap



Memory Diagram Walkthrough: 10

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }
    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }
    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

Stack

BankAccount(
String,int,int)

Param				
balance	50000	number	101	name
Local				
Member				
balance	0	acctNum	0	name

main: 3

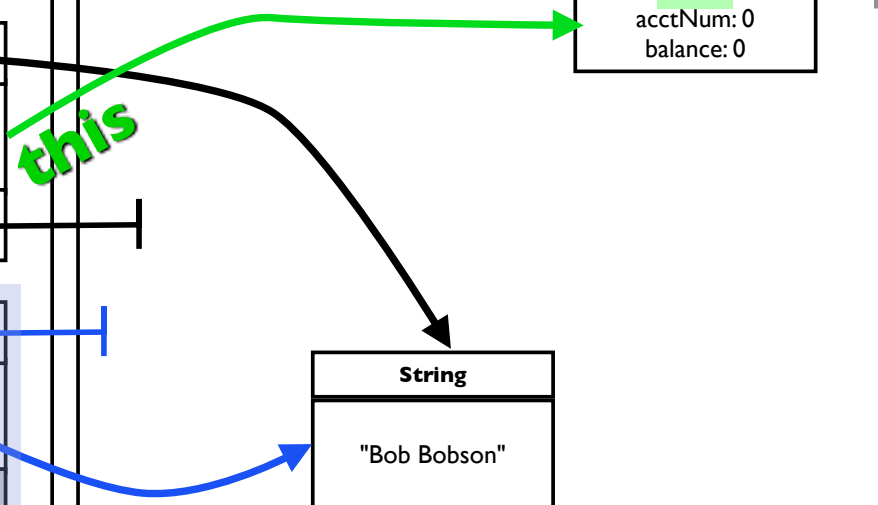
Param		args
Local		name
startBalance	50000	
Member		

Heap

BankAccount	
name:	
acctNum:	0
balance:	0

String	
"Bob Bobson"	

this



Memory Diagram Walkthrough: II

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

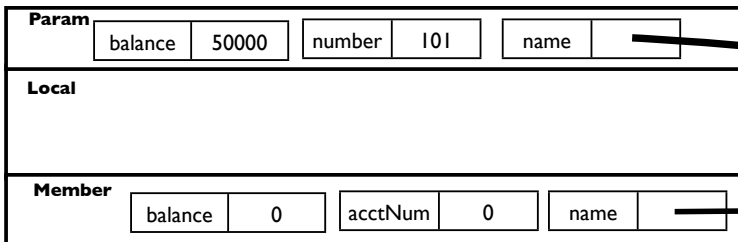
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack

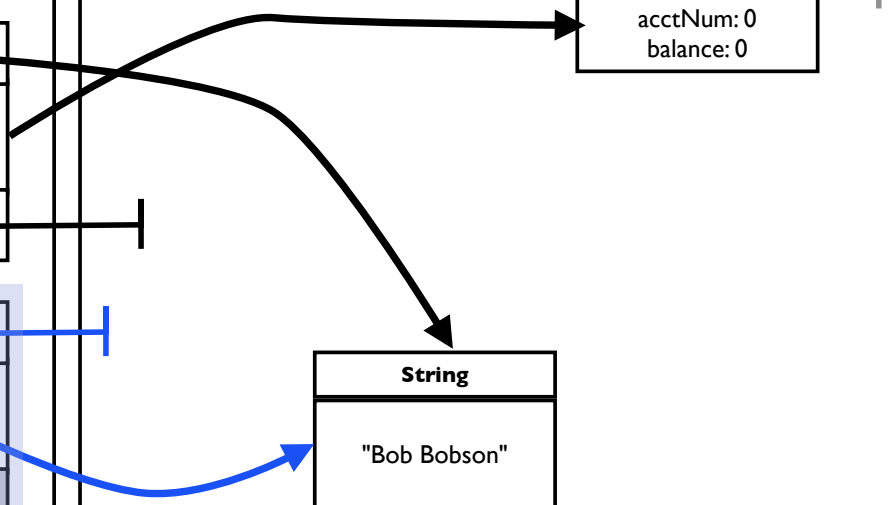
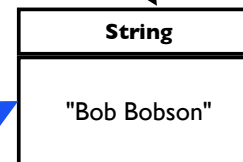
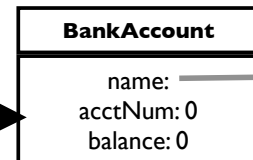
BankAccount(
String,int,int)



main: 3



Heap



Memory Diagram Walkthrough: 12

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }

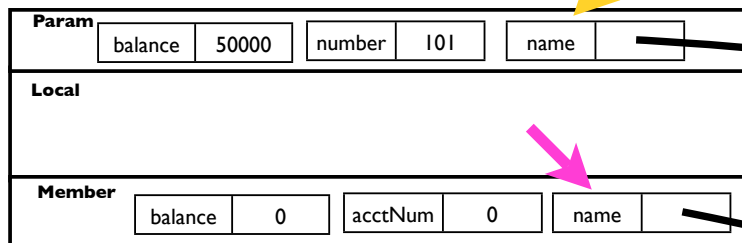
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

Stack

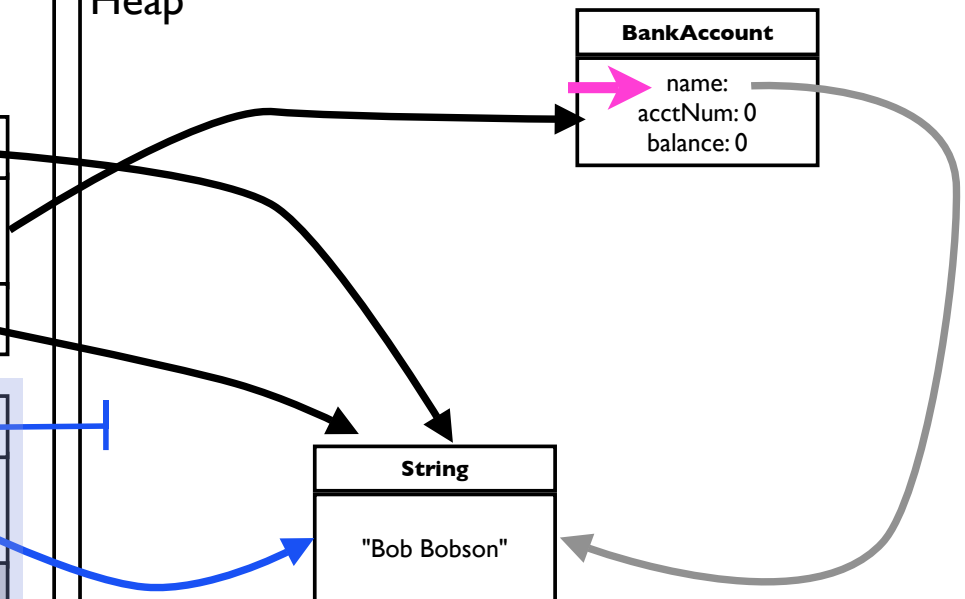
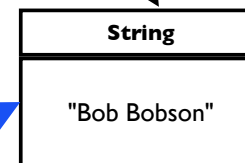
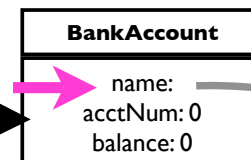
BankAccount(
String,int,int)



main: 3



Heap



Memory Diagram Walkthrough: 13

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

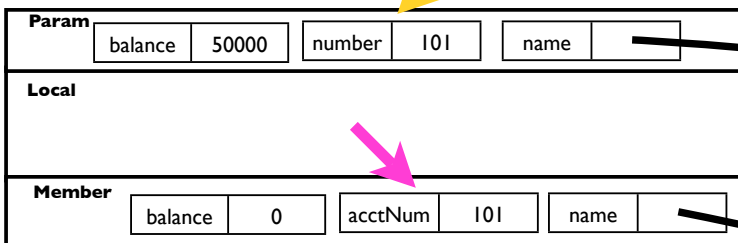
BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }
    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }
    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

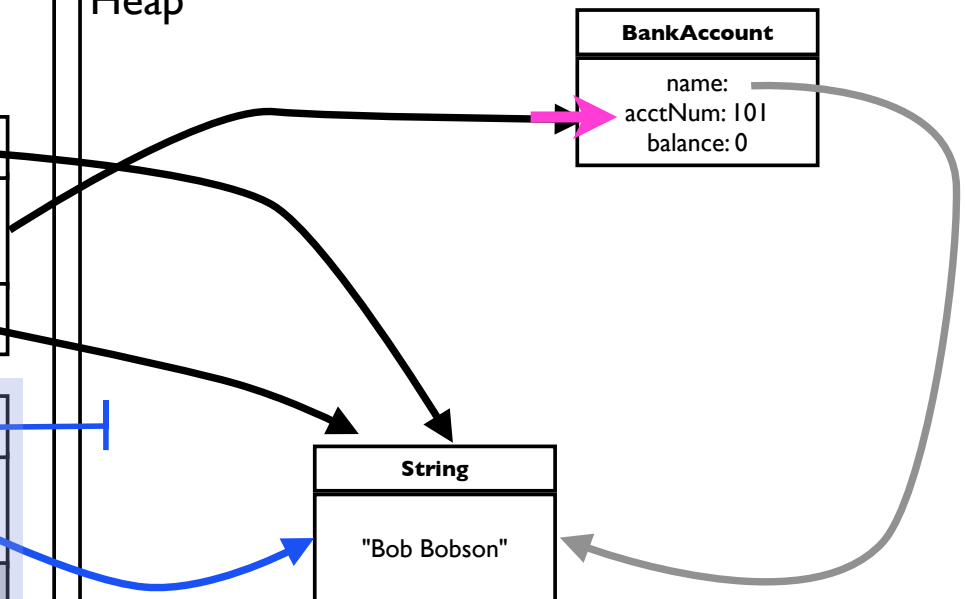
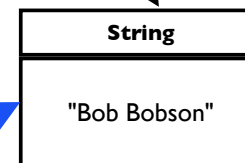
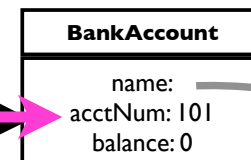
Stack

BankAccount(
String,int,int)

main: 3



Heap



Memory Diagram Walkthrough: 14

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

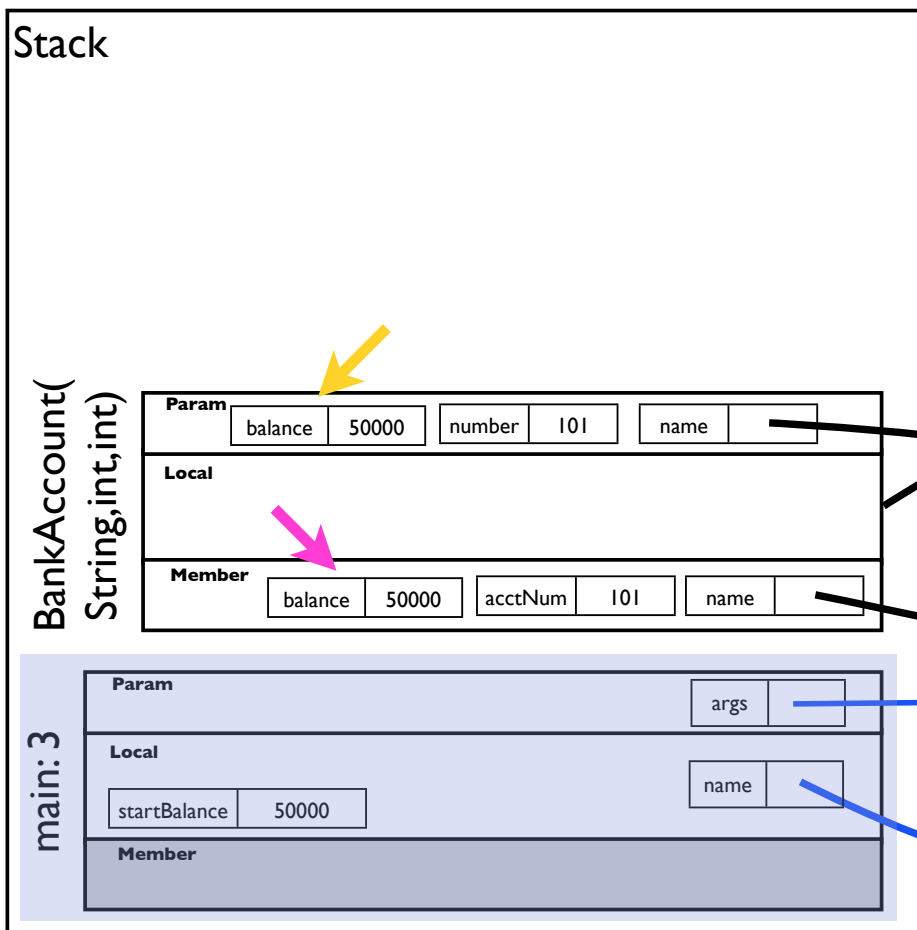
    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

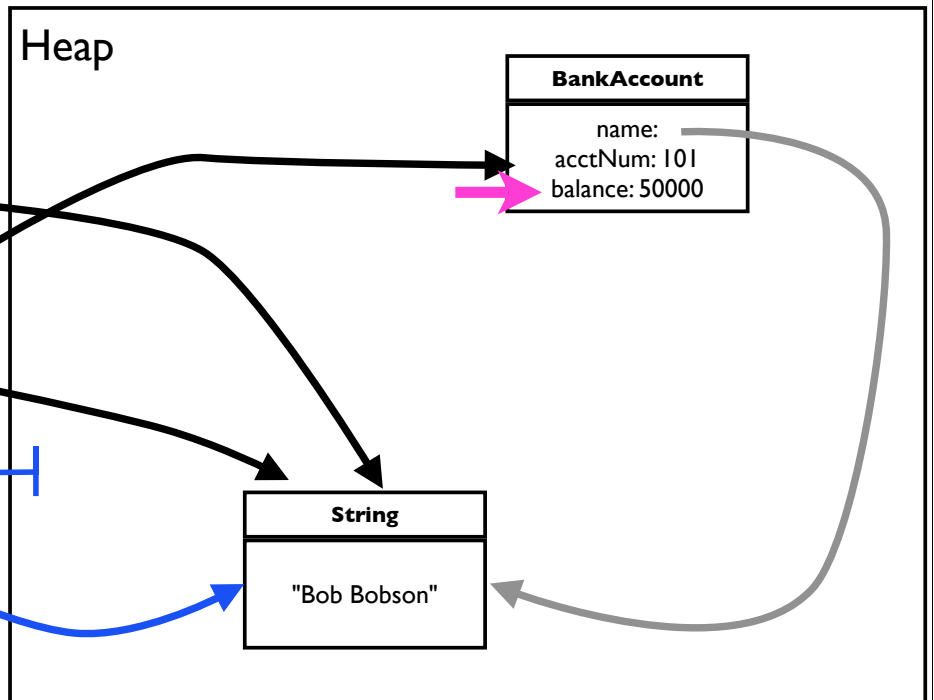
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack



Heap



Memory Diagram Walkthrough: 15

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

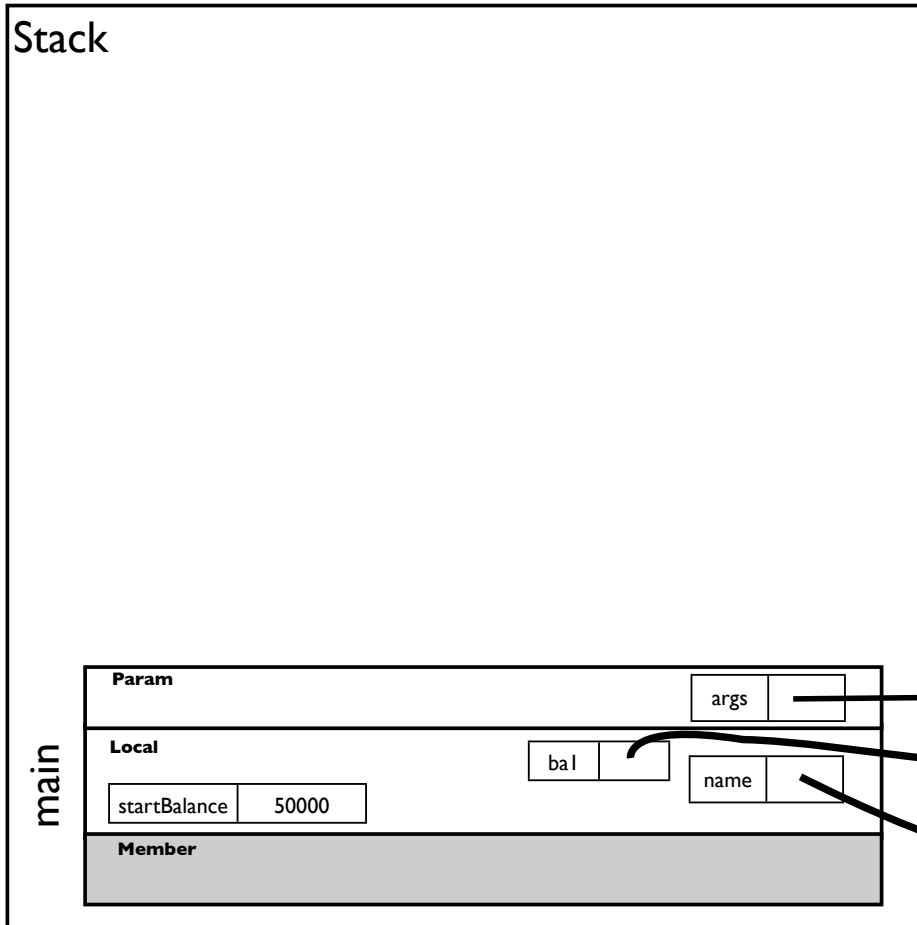
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

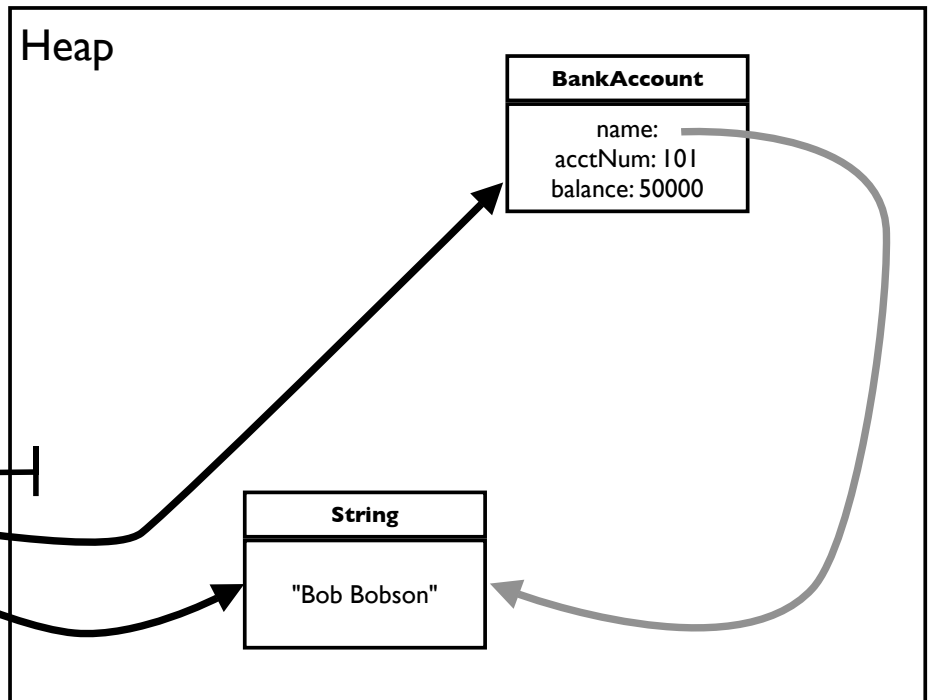
    private void setBalance(int newAmount) {    balance = newAmount;    }

    public int getBalance() {    return balance;    }
}
```

Stack



Heap



Memory Diagram Walkthrough: 16

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) {    balance = newAmount;    }

    public int getBalance() {    return balance;    }
}
```

Stack

makeDeposit

Param	
amount	20000
Local	
Member	
balance	50000
acctNum	101
name	

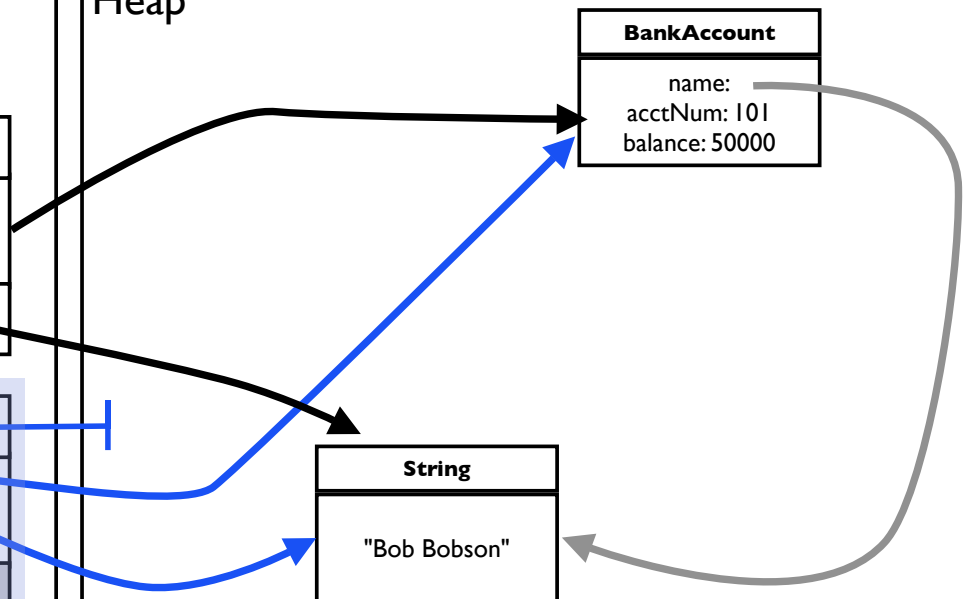
main: 4

Param	
args	
Local	
startBalance	50000
ba1	
name	
Member	

Heap

BankAccount	
name:	
acctNum:	101
balance:	50000

String	
"Bob Bobson"	



Memory Diagram Walkthrough: 17

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack

makeDeposit

Param	
amount	20000
Local	
Member	
balance	50000
acctNum	101
name	

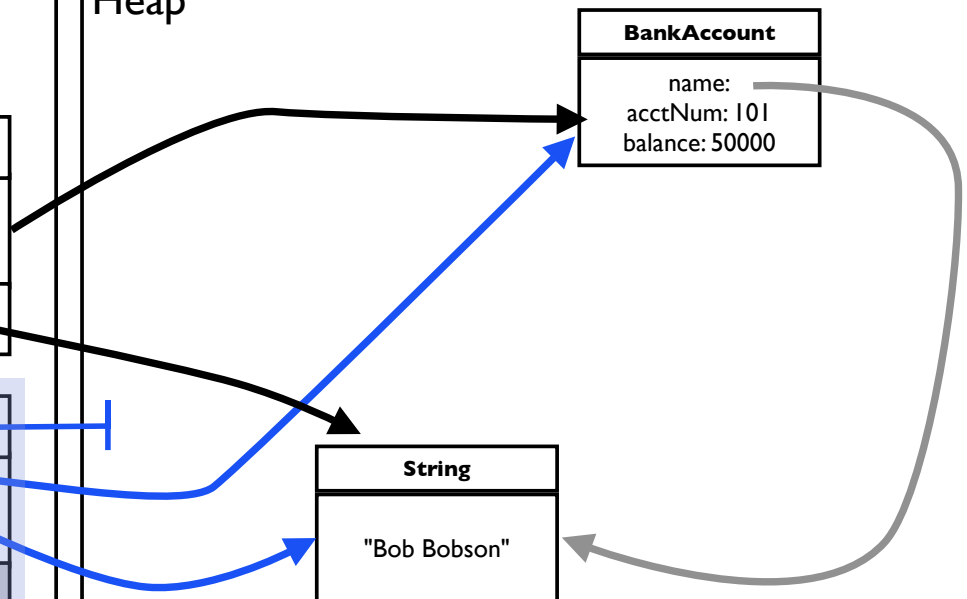
main: 4

Param	
args	
Local	
startBalance	50000
ba1	
name	
Member	

Heap

BankAccount	
name:	
acctNum: 101	
balance: 50000	

String	
"Bob Bobson"	



Memory Diagram Walkthrough: 18

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

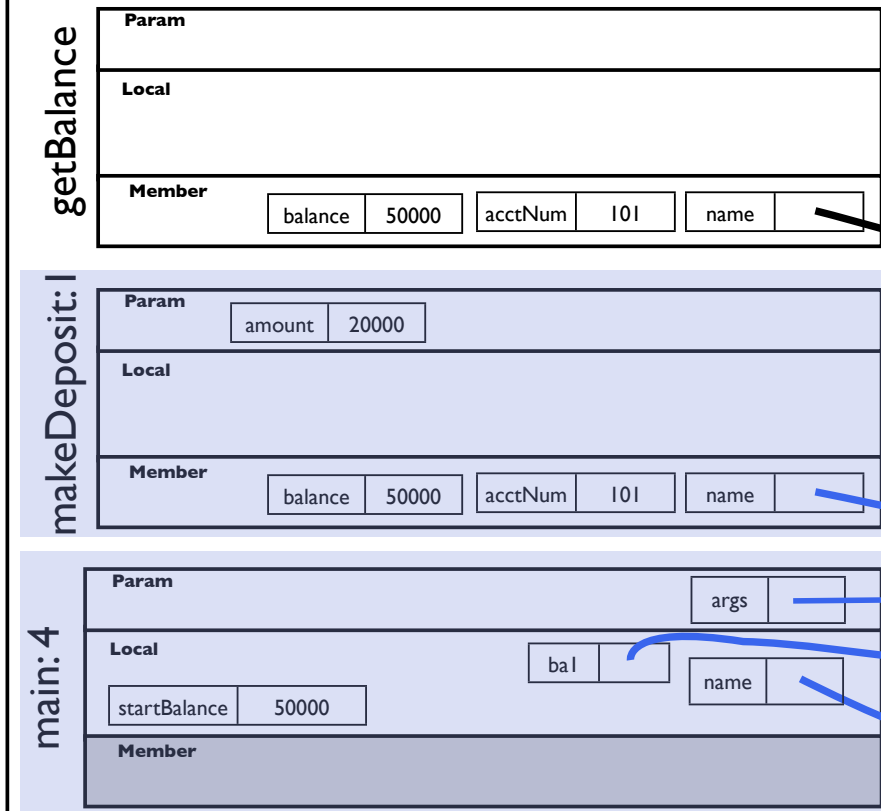
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

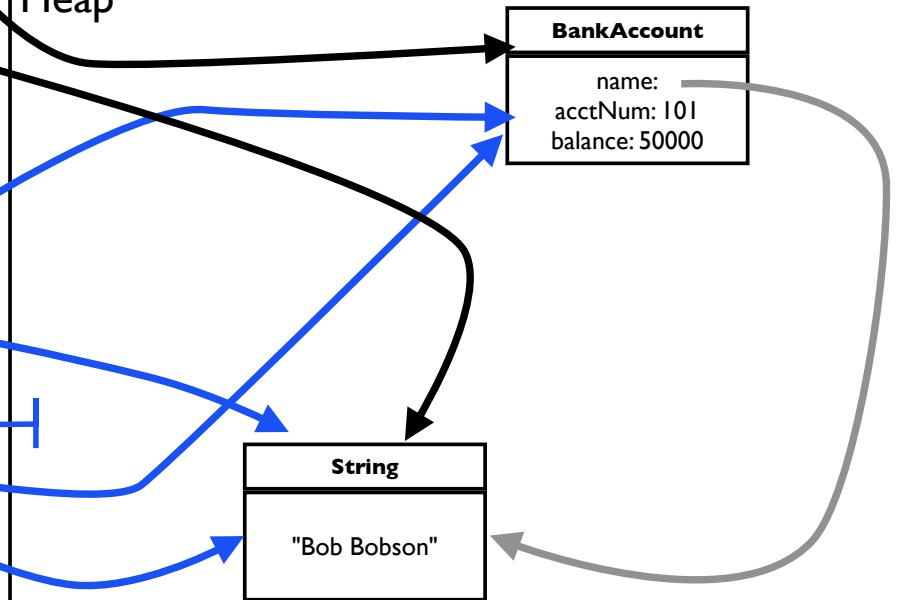
    private void setBalance(int newAmount) {    balance = newAmount;    }

    public int getBalance() {    return balance;    }
}
```

Stack



Heap



Memory Diagram Walkthrough: 19

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack

makeDeposit

Param	
amount	20000
Local	
Member	
balance	50000
acctNum	101
name	

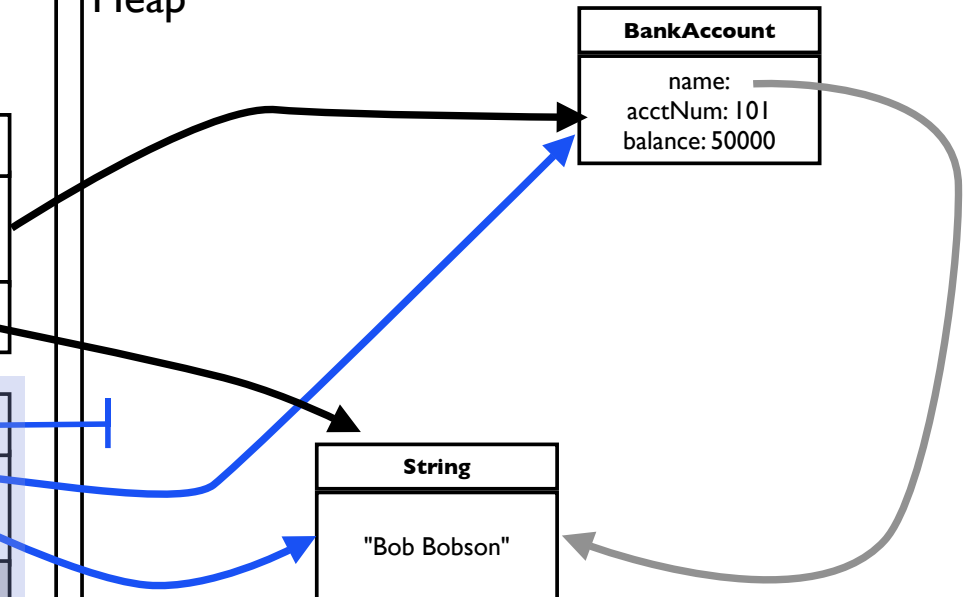
main: 4

Param	
args	
Local	
startBalance	50000
ba1	
name	
Member	

Heap

BankAccount	
name:	
acctNum: 101	
balance: 50000	

String	
"Bob Bobson"	



Memory Diagram Walkthrough: 20

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(n          50000          + 20000          = 70000
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack

makeDeposit

Param	
amount	20000
Local	
Member	
balance	50000
acctNum	101
name	

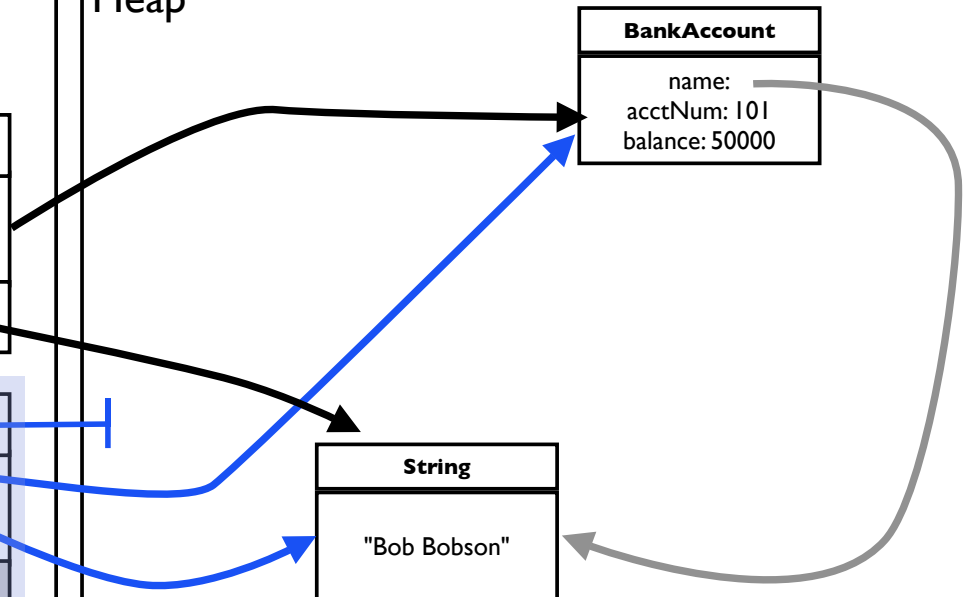
main: 4

Param	
args	
Local	
startBalance	50000
ba1	
name	
Member	

Heap

BankAccount	
name:	
acctNum: 101	
balance: 50000	

String	
"Bob Bobson"	



Memory Diagram Walkthrough: 21

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack

makeDeposit

Param	amount	20000
Local	newBalance	70000
Member	balance	50000
	acctNum	101
	name	

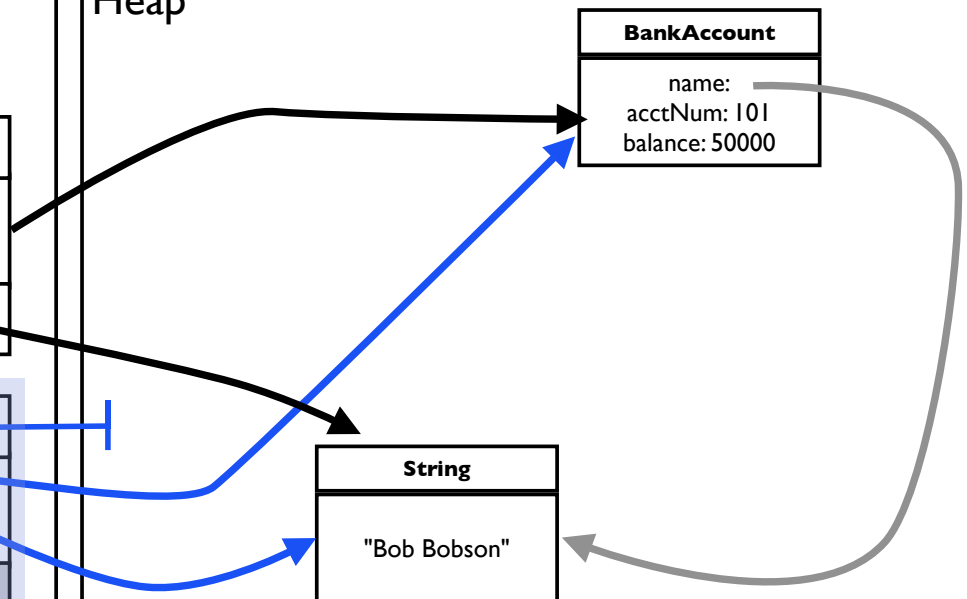
main: 4

Param	args	
Local	startBalance	50000
	ba1	
	name	
Member		

Heap

BankAccount
name:
acctNum: 101
balance: 50000

String
"Bob Bobson"



Memory Diagram Walkthrough: 22

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

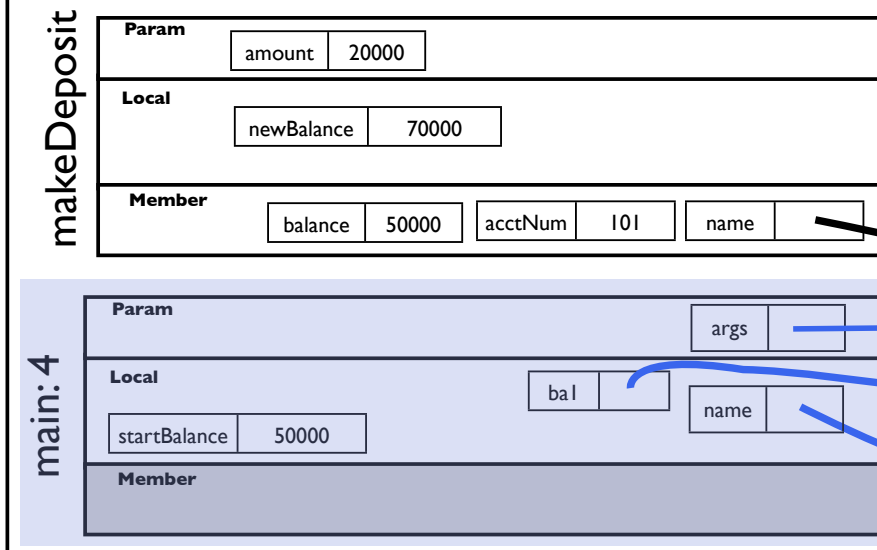
    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

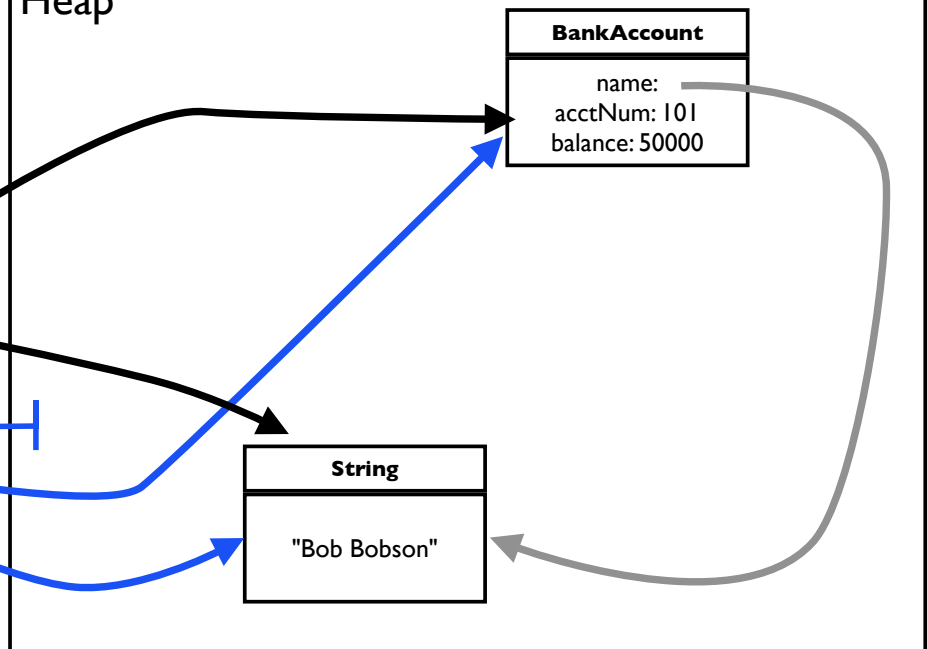
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack



Heap



Memory Diagram Walkthrough: 23

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

Stack

setBalance

Param	newAmount	70000
Local		
Member	balance	50000
	acctNum	101
	name	

makeDeposit:2

Param	amount	20000
Local	newBalance	70000
Member	balance	50000
	acctNum	101
	name	

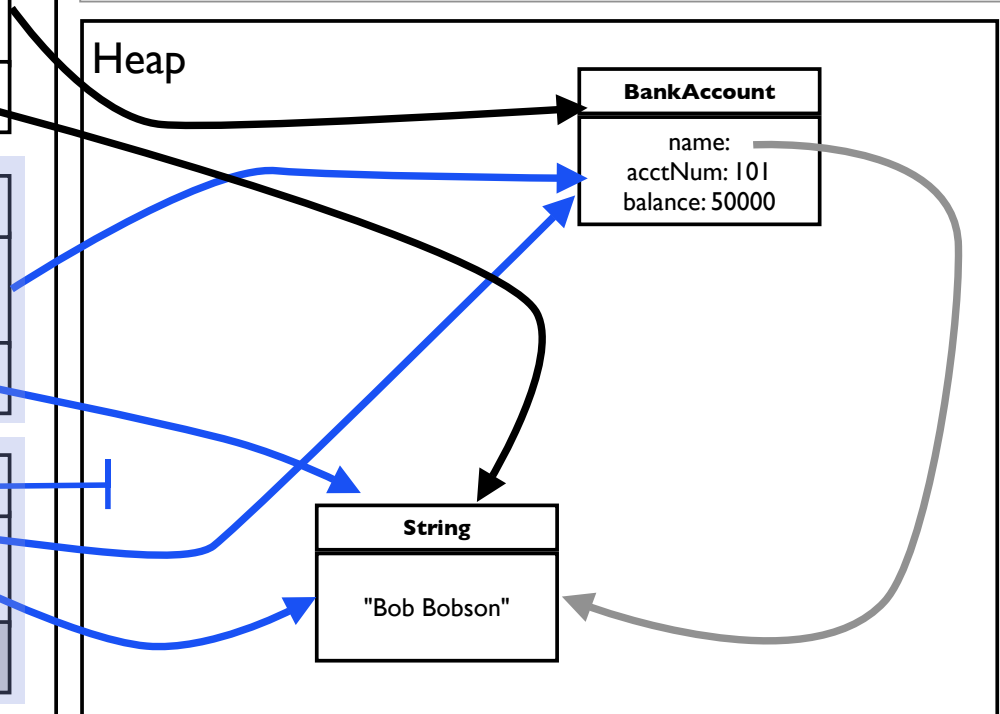
main: 4

Param		args	
Local	startBalance	50000	
	ba1		
	name		
Member			

Heap

BankAccount
name:
acctNum: 101
balance: 50000

String
"Bob Bobson"



Memory Diagram Walkthrough: 24

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

Stack

setBalance

Param	newAmount	70000
Local		
Member	balance	70000
	acctNum	101
	name	

makeDeposit:2

Param	amount	20000
Local	newBalance	70000
Member	balance	50000
	acctNum	101
	name	

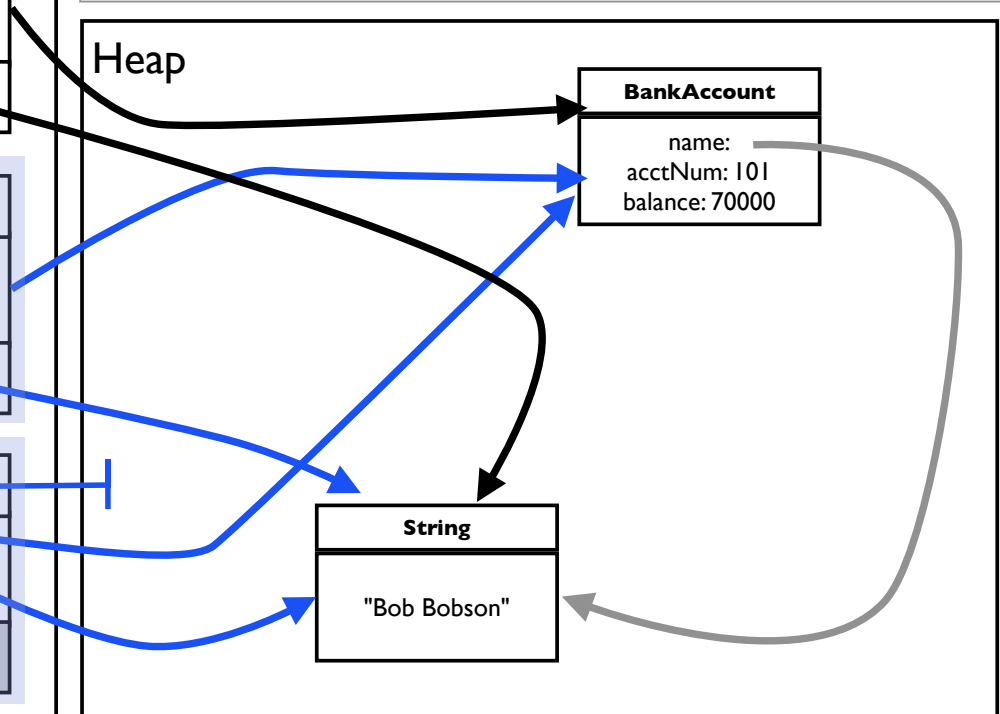
main: 4

Param	args	
Local	startBalance	50000
	ba1	
	name	
Member		

Heap

BankAccount
name:
acctNum: 101
balance: 70000

String
"Bob Bobson"



Memory Diagram Walkthrough: 25

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack

makeDeposit

Param	amount	20000
Local	newBalance	70000
Member	balance	70000
	acctNum	101
	name	

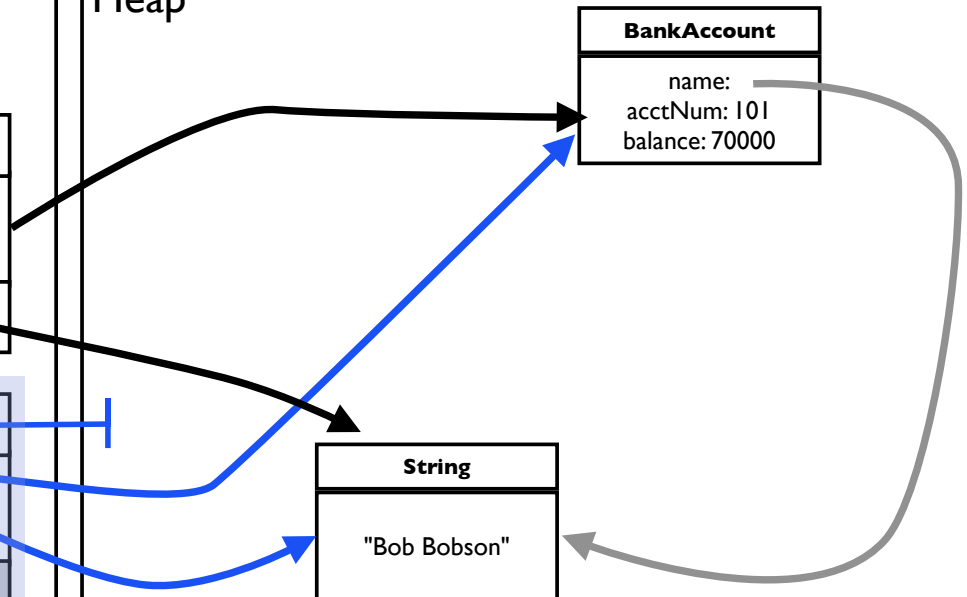
main: 4

Param	args
Local	startBalance: 50000
	ba1
	name
Member	

Heap

BankAccount
name:
acctNum: 101
balance: 70000

String
"Bob Bobson"



Memory Diagram Walkthrough: 26

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack

makeDeposit

Param		amount	20000
Local		newBalance	70000
Member		balance	70000
		acctNum	101
		name	

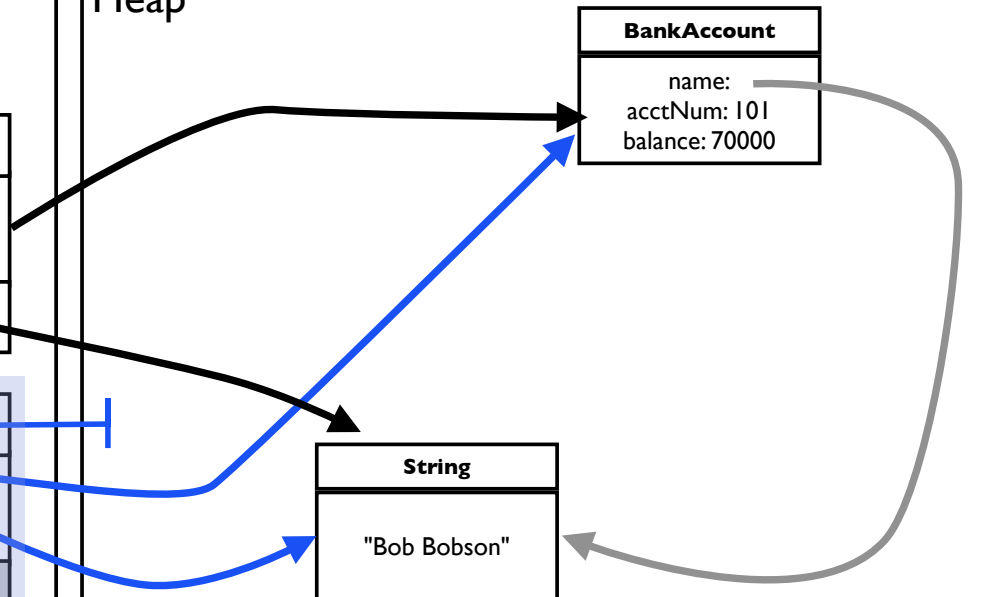
main: 4

Param		args	
Local		startBalance	50000
		ba1	
		name	
Member			

Heap

BankAccount	
name:	
acctNum:	101
balance:	70000

String	
"Bob Bobson"	



Memory Diagram Walkthrough: 27

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

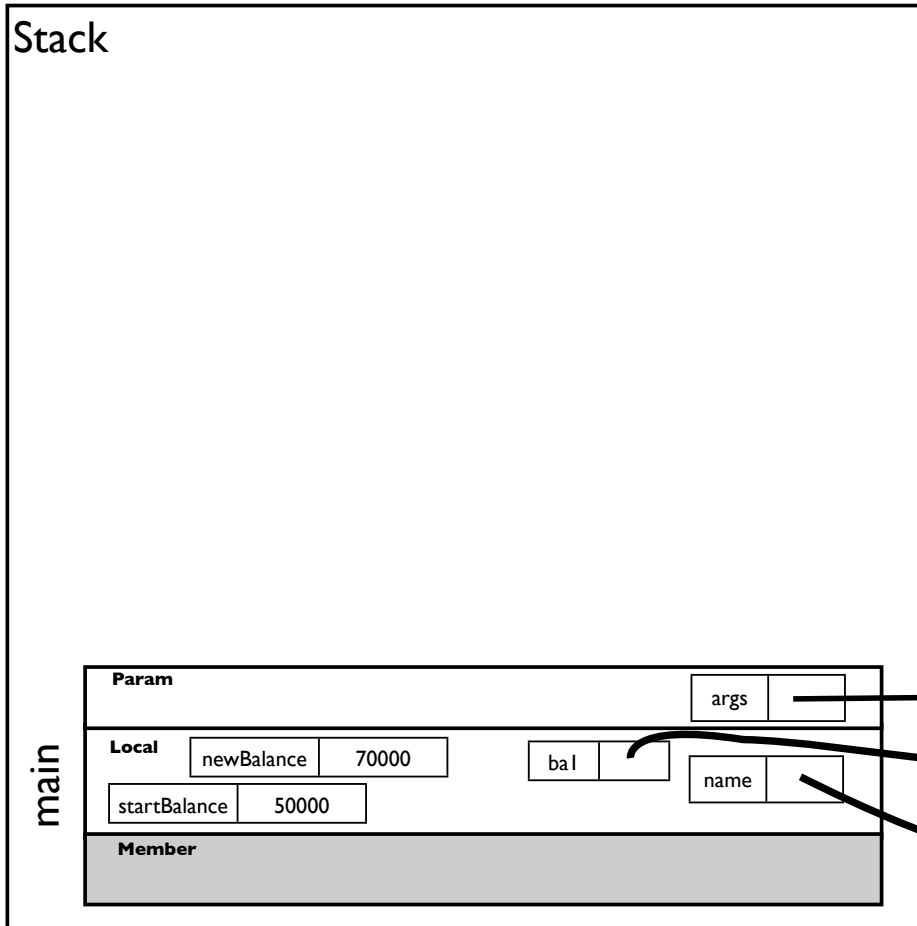
    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

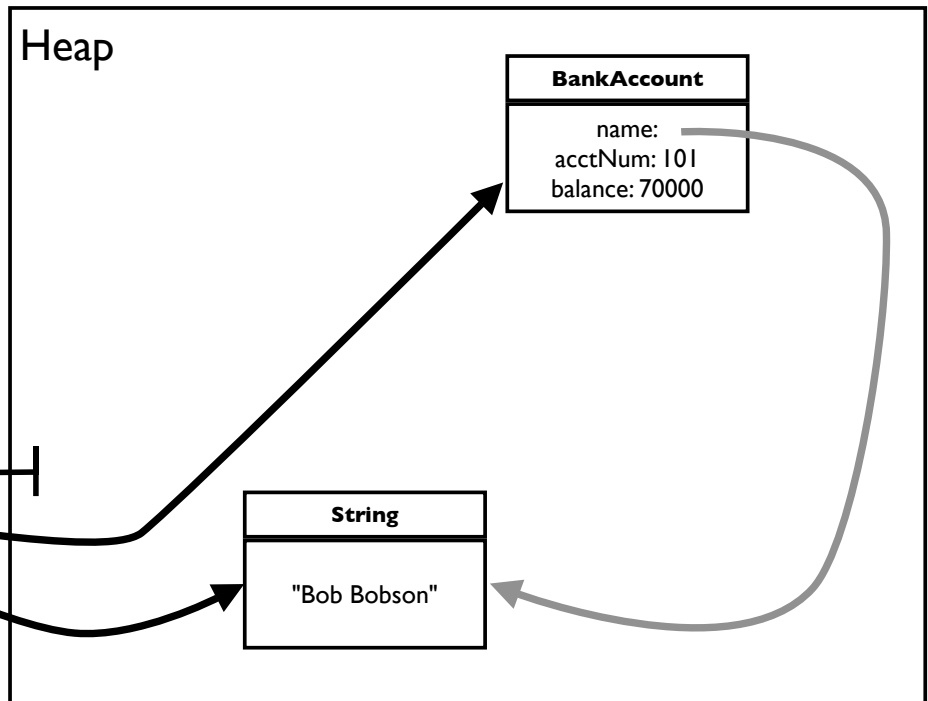
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack



Heap



Memory Diagram Walkthrough: 28

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

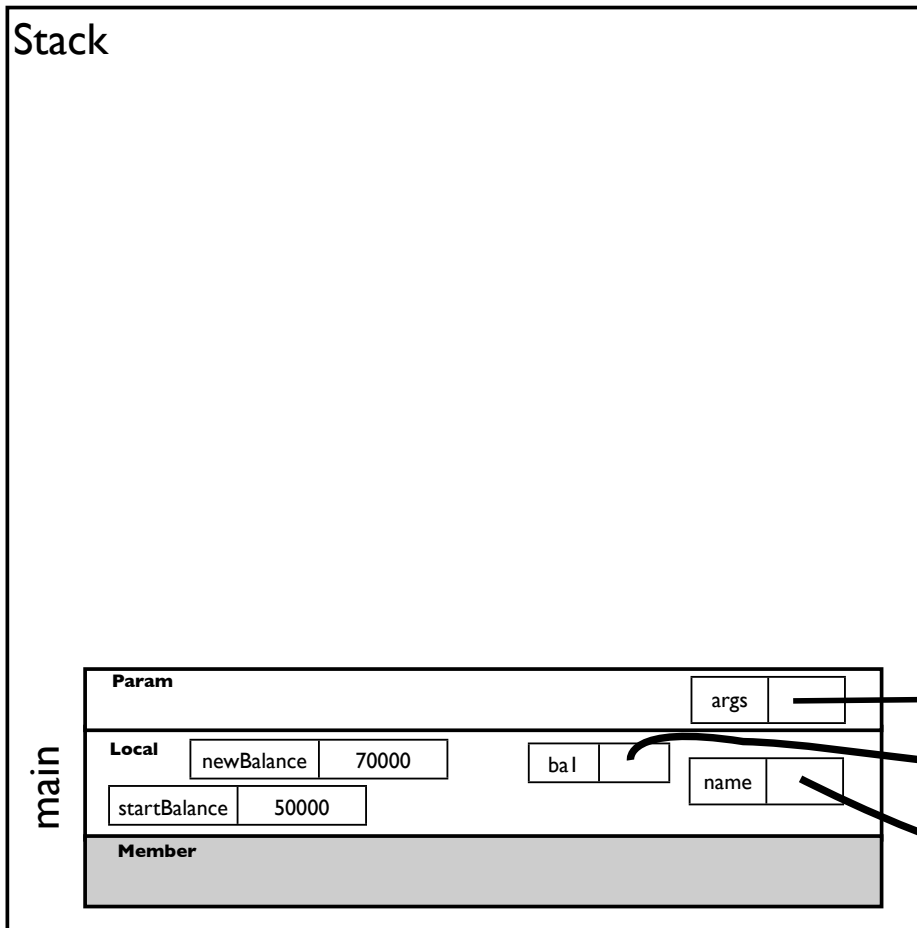
    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

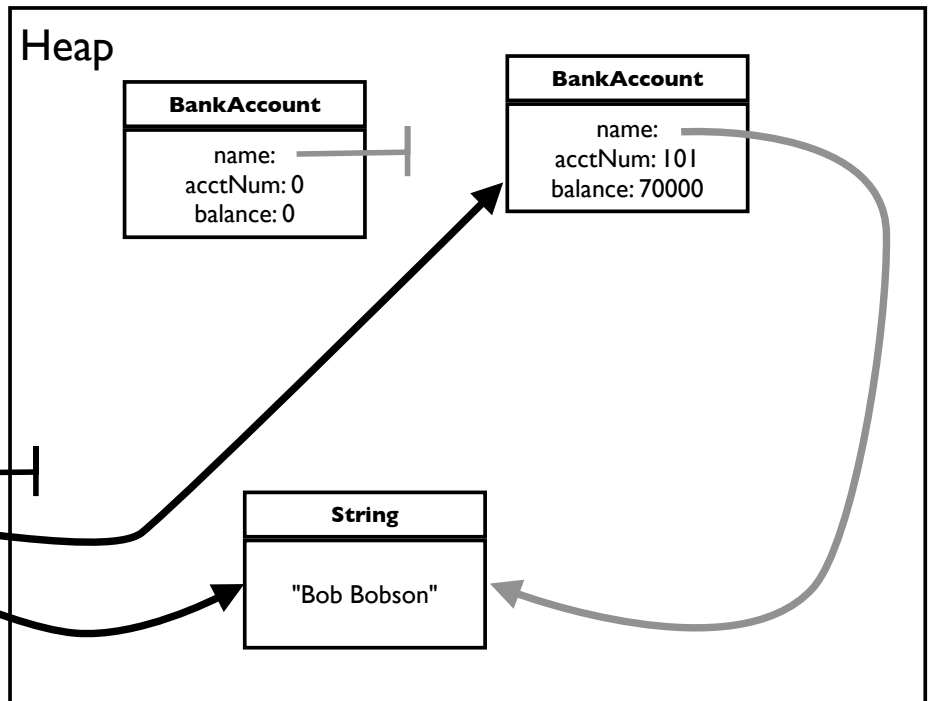
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack



Heap



Memory Diagram Walkthrough: 29

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

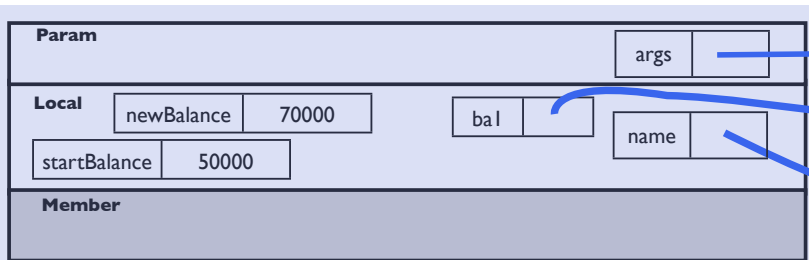
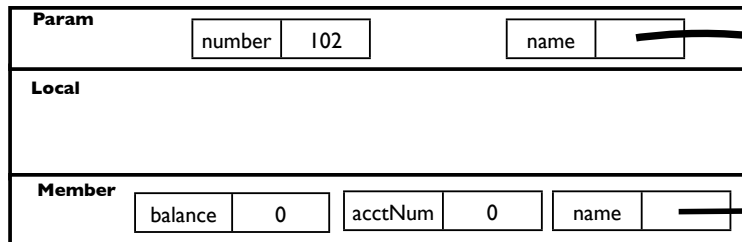
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

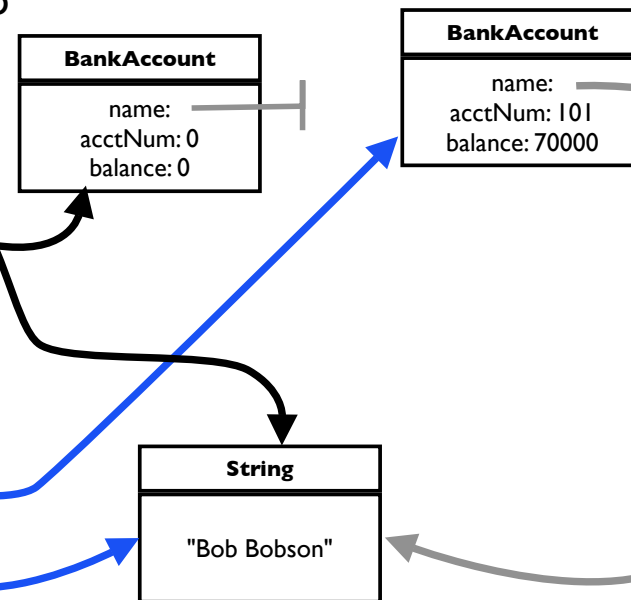
Stack

BankAccount(
String,int)

main: 5



Heap



Memory Diagram Walkthrough: 30

BankApplication.java

```
public class BankApplication {  
    public static void main(String[] args) {  
        String name = "Bob Bobson";  
        int startBalance = 50000;  
        BankAccount ba1 = new BankAccount(name, 101, startBalance);  
        int newBalance = ba1.makeDeposit(20000);  
        BankAccount ba2 = new BankAccount(name, 102);  
        ba1.transferMoney(ba2, 15000);  
    }  
}
```

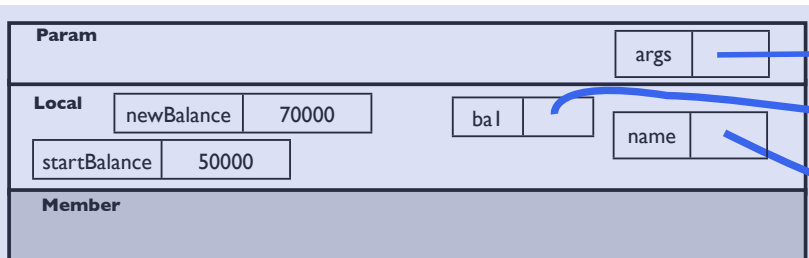
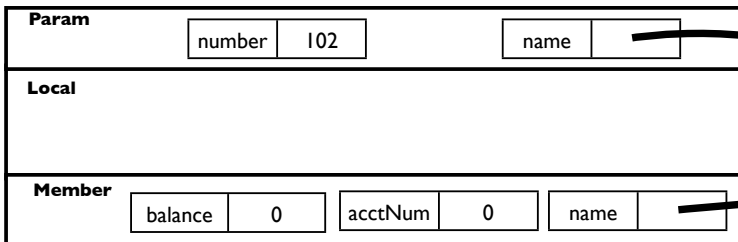
BankAccount.java

```
public class BankAccount {  
    public BankAccount(String name, int number) {  
        this.name = name;    this.acctNum = number;    this.balance = 0;  
    }  
  
    public BankAccount(String name, int number, int balance) {  
        this.name = name;    this.acctNum = number;    this.balance = balance;  
    }  
  
    public int makeDeposit(int amount) {  
        int newBalance = this.getBalance() + amount;  
        this.setBalance(newBalance);  
        return newBalance;  
    }  
  
    public void transferMoney(BankAccount b, int amount) {  
        this.setBalance(this.getBalance() - amount);  
        b.setBalance(b.getBalance() + amount);  
    }  
  
    private void setBalance(int newAmount) {    balance = newAmount;    }  
    public int getBalance() {    return balance;    }  
}
```

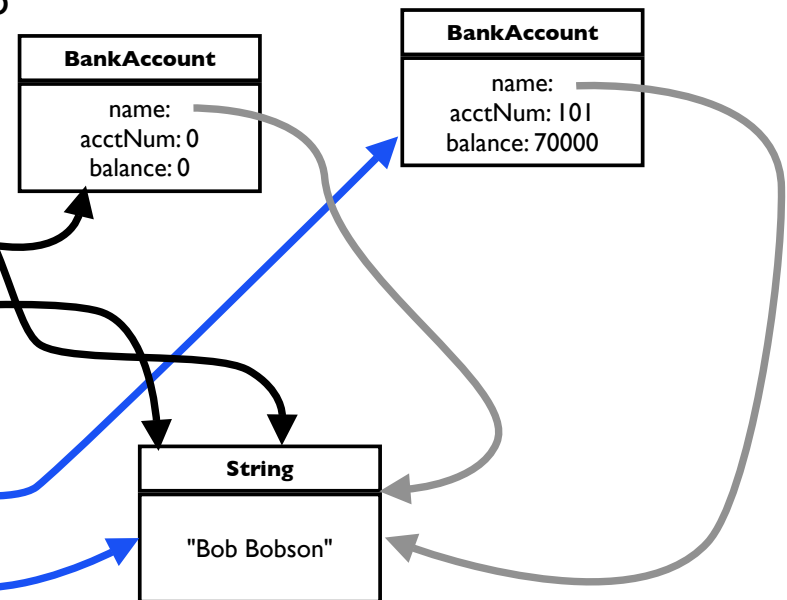
Stack

BankAccount(
String,int)

main: 5



Heap



Memory Diagram Walkthrough: 31

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

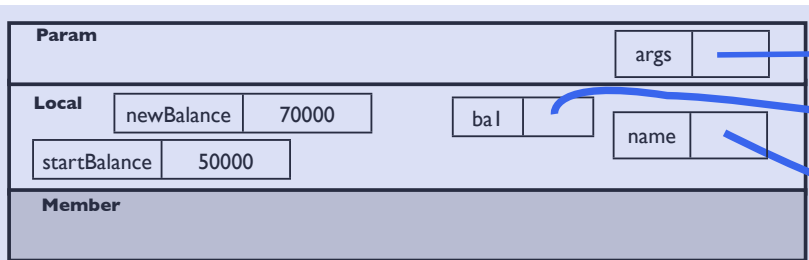
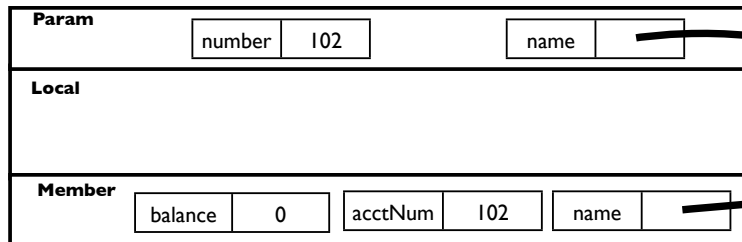
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

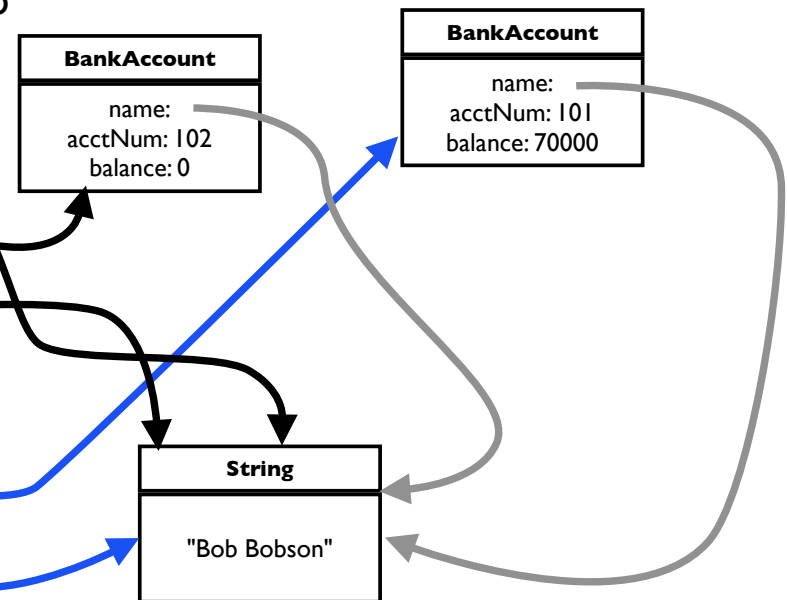
Stack

BankAccount(
String,int)

main: 5



Heap



Memory Diagram Walkthrough: 32

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }

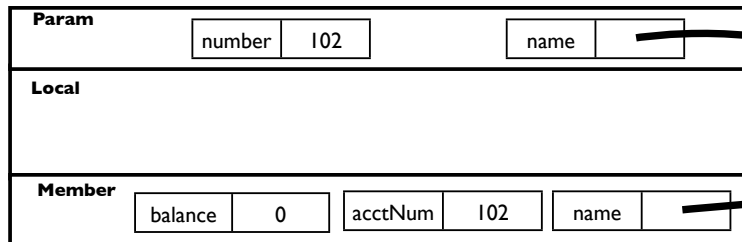
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

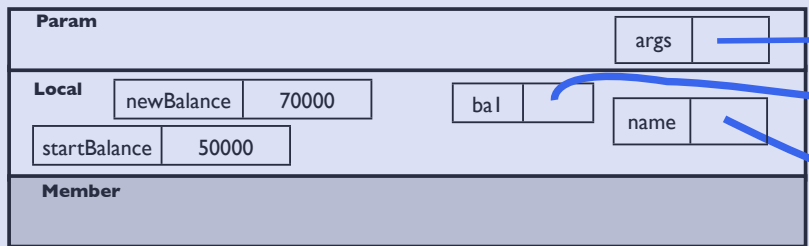
    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

Stack

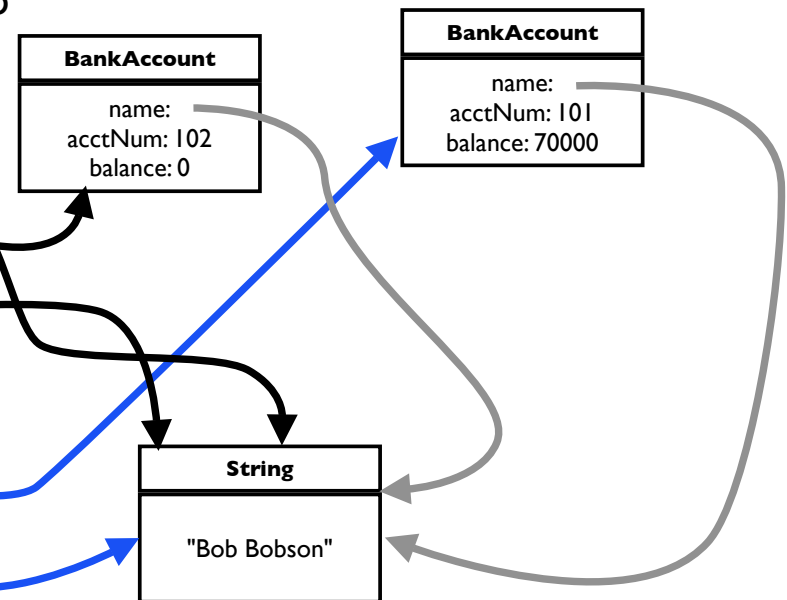
BankAccount(
String,int)



main: 5



Heap



Memory Diagram Walkthrough: 33

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

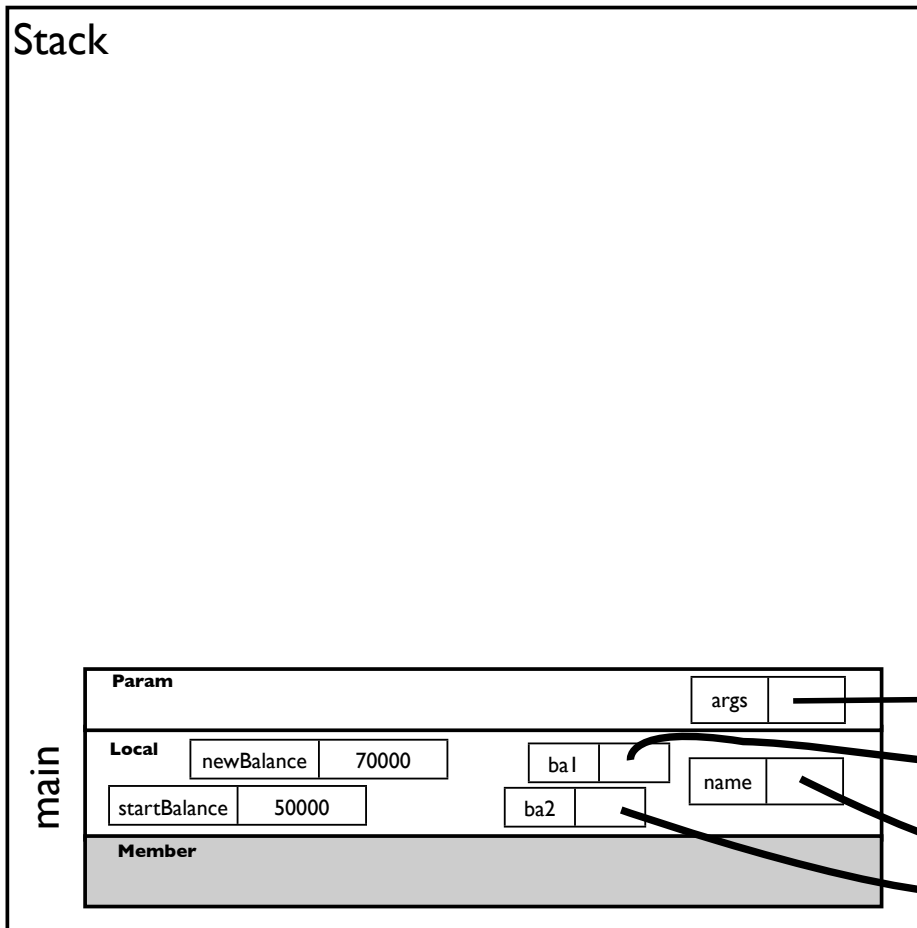
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

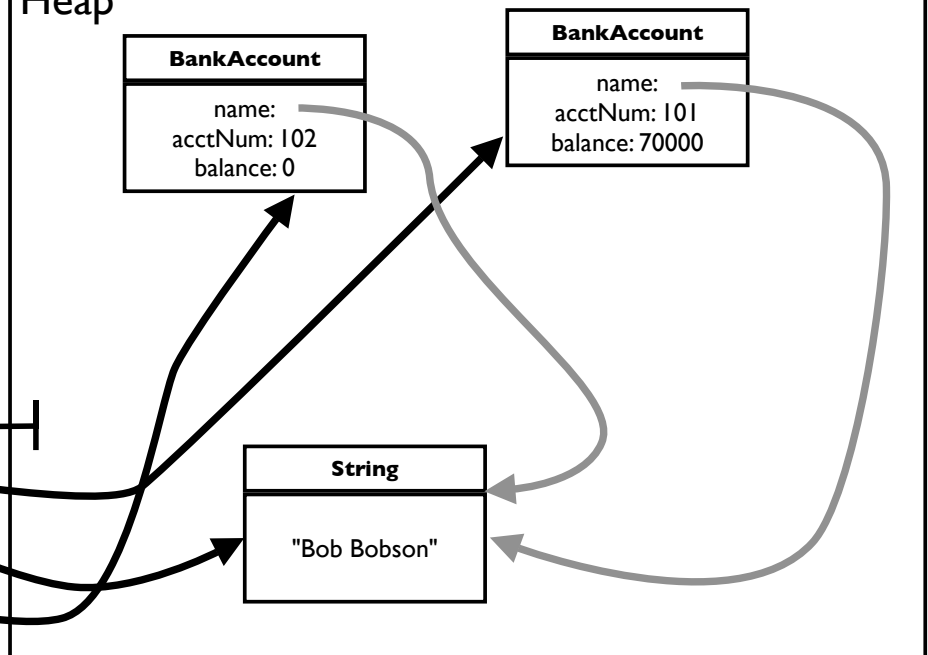
    private void setBalance(int newAmount) {    balance = newAmount;    }

    public int getBalance() {    return balance;    }
}
```

Stack



Heap



Memory Diagram Walkthrough: 34

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

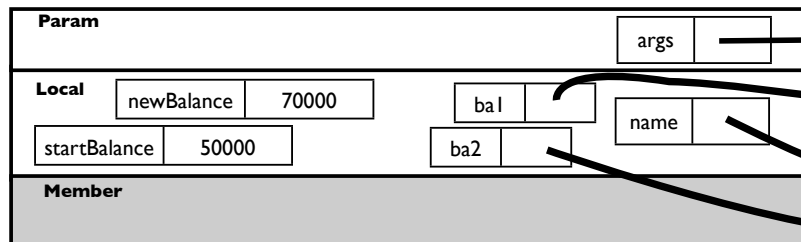
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

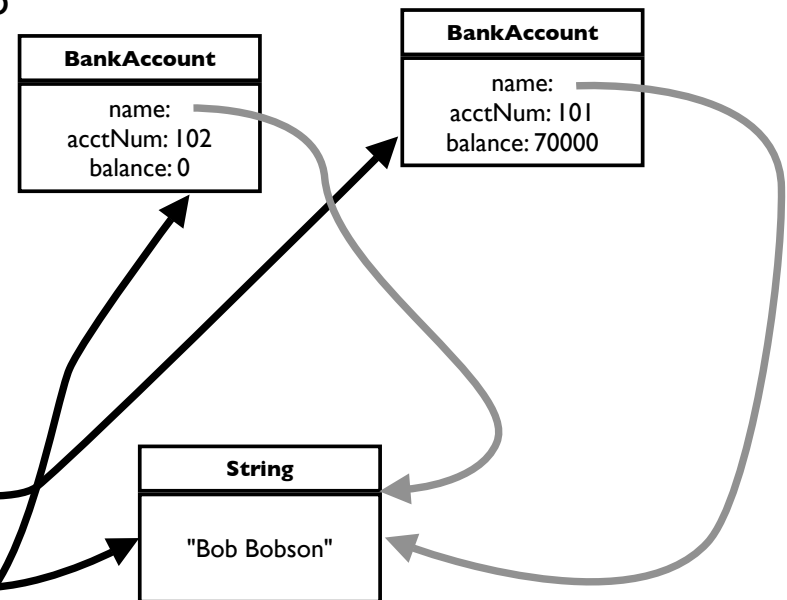
    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack

main: 6



Heap



Memory Diagram Walkthrough: 35

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }

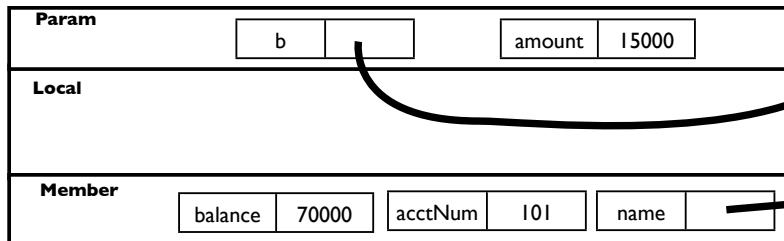
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

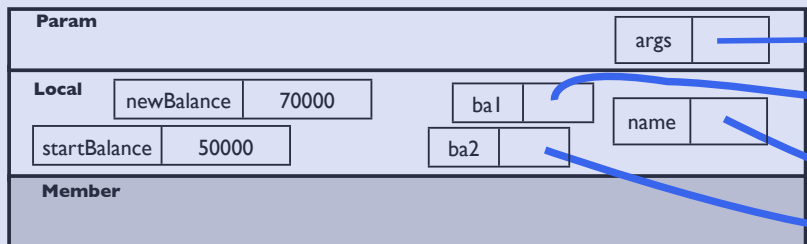
    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

Stack

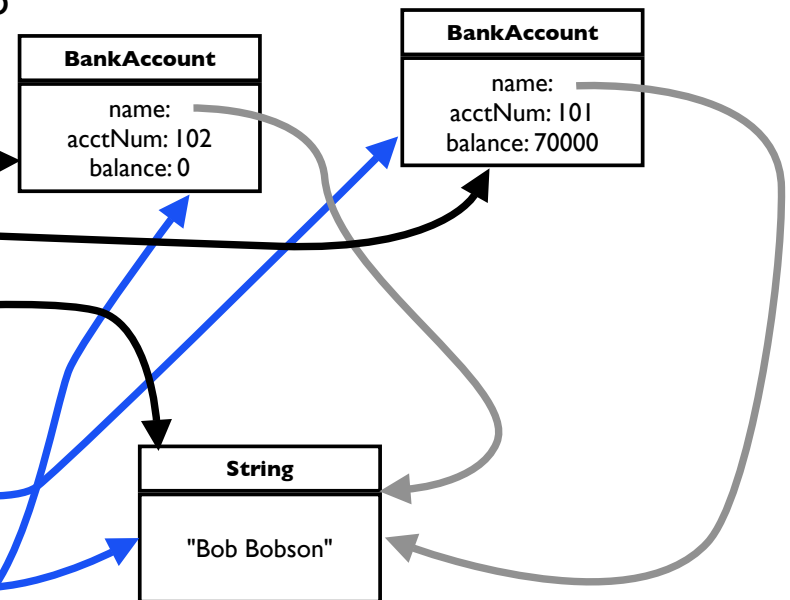
transferMoney



main: 6



Heap



Memory Diagram Walkthrough: 36

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }

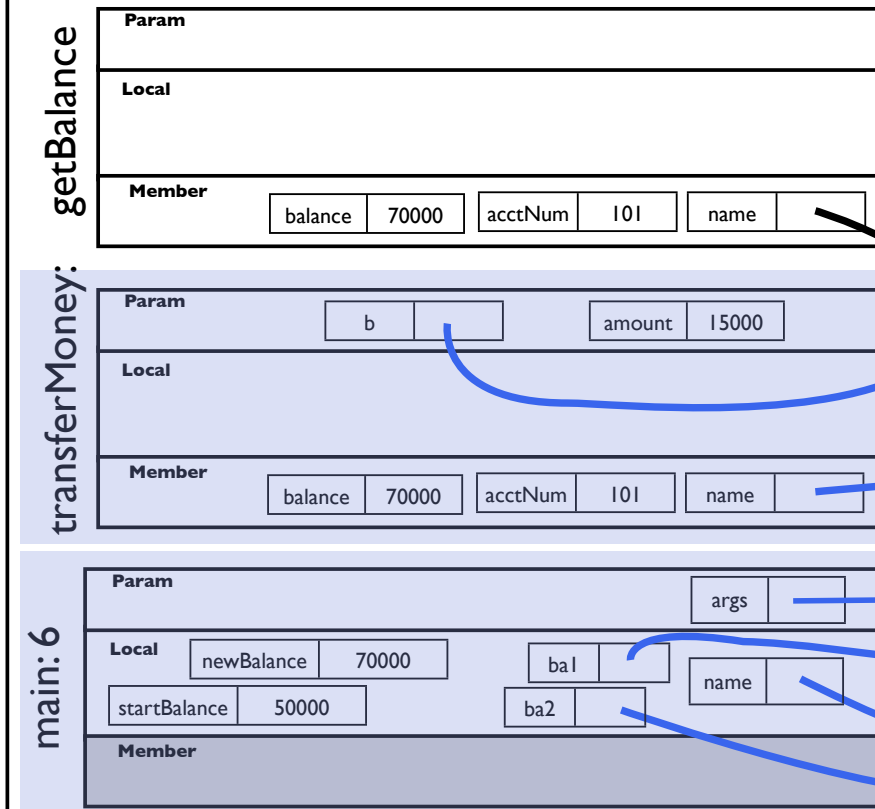
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

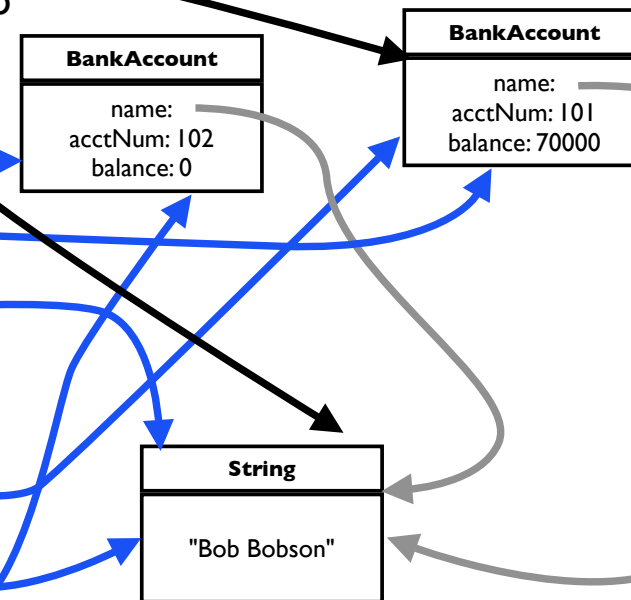
    private void setBalance(int newAmount) { balance = newAmount; }

    public int getBalance() { return balance; }
}
```

Stack



Heap



Memory Diagram Walkthrough: 37

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

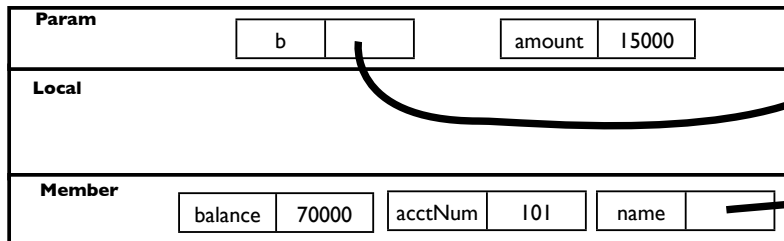
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

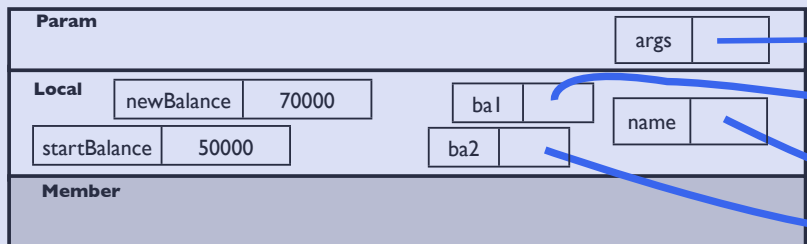
    private void setBalance(int newAmount) {    balance = newAmount;    }
    public int getBalance() {    return balance;    }
}
```

Stack

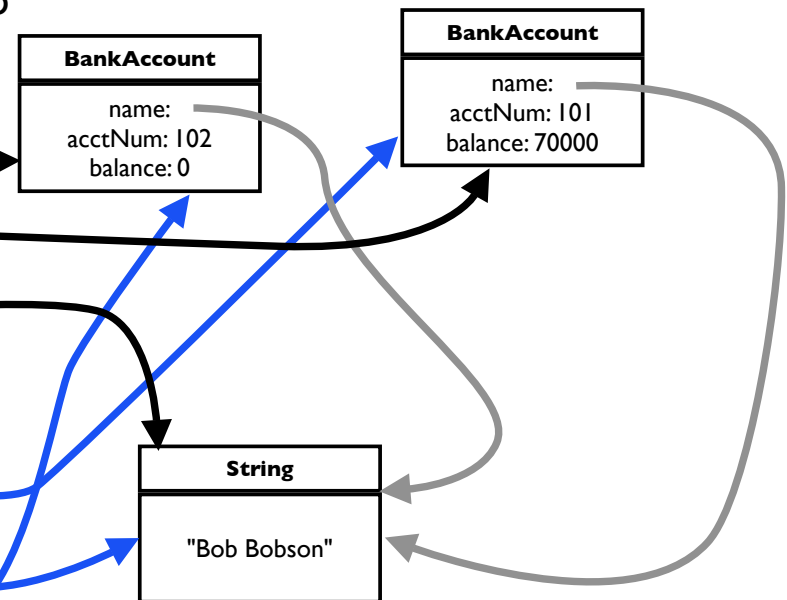
transferMoney



main: 6



Heap



Memory Diagram Walkthrough: 38

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }

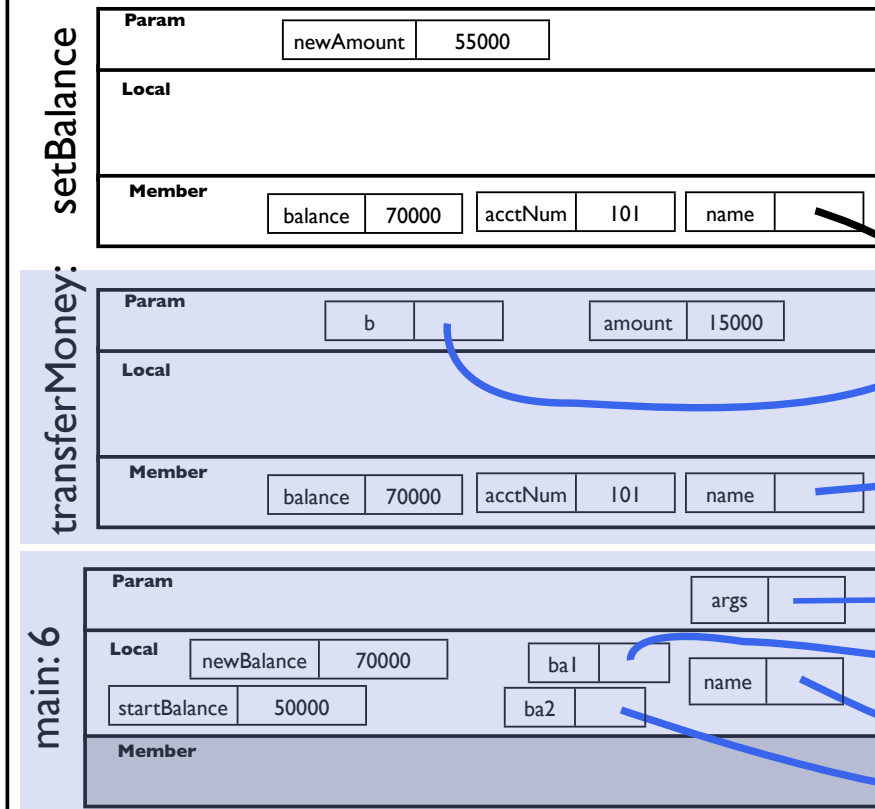
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

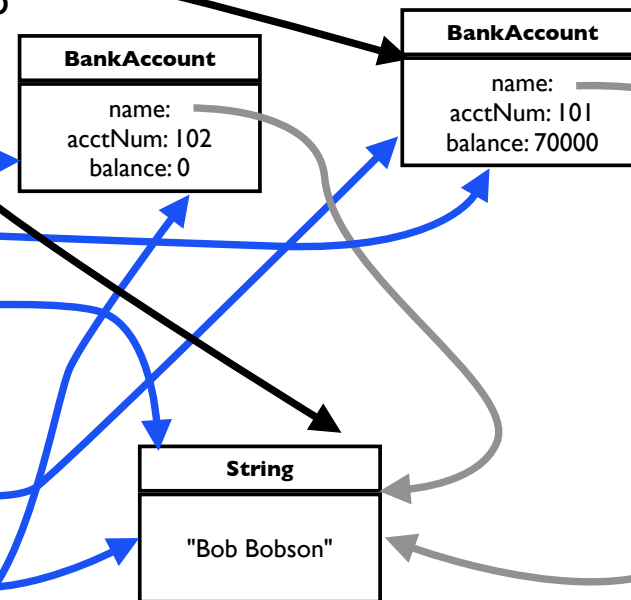
    private void setBalance(int newAmount) { balance = newAmount; }

    public int getBalance() { return balance; }
}
```

Stack



Heap



Memory Diagram Walkthrough: 39

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }

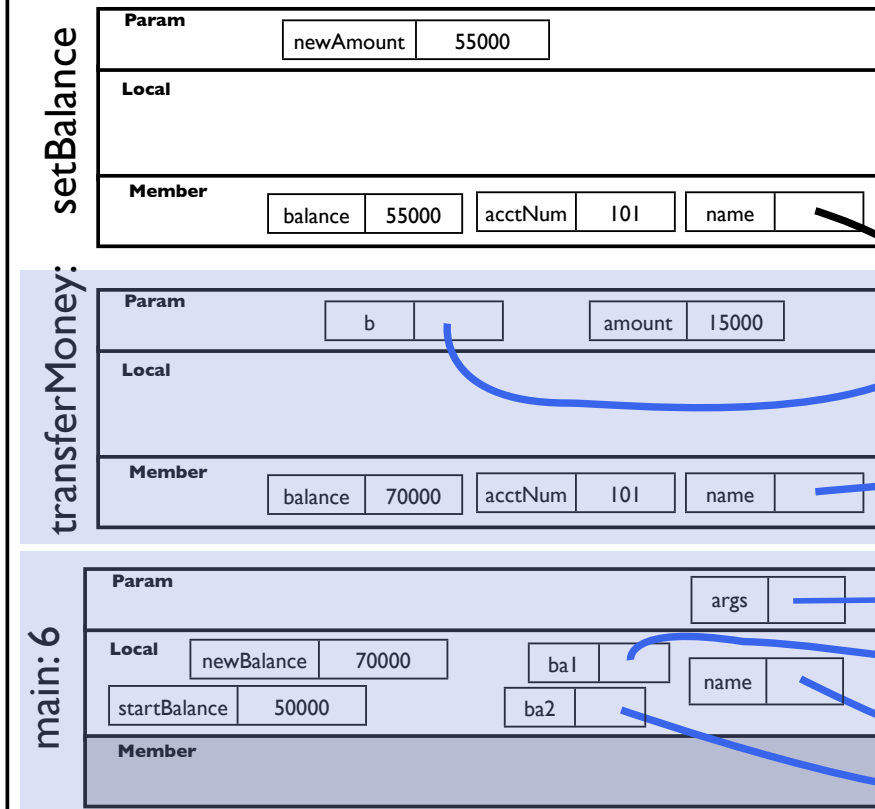
    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

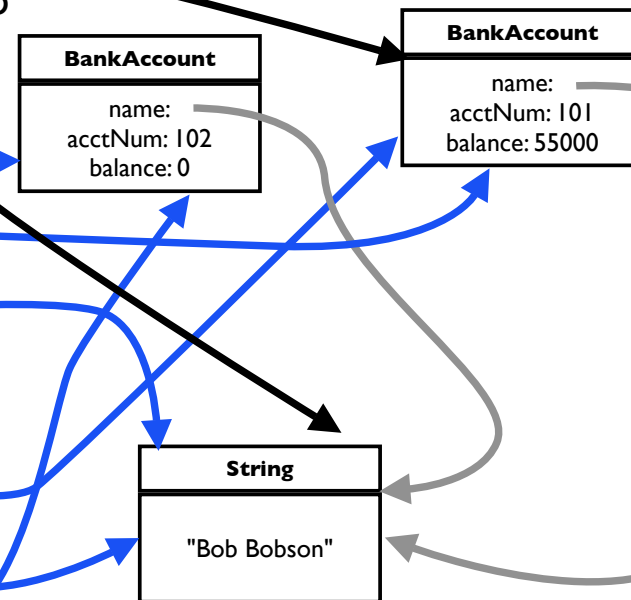
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

Stack



Heap



Memory Diagram Walkthrough: 40

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }

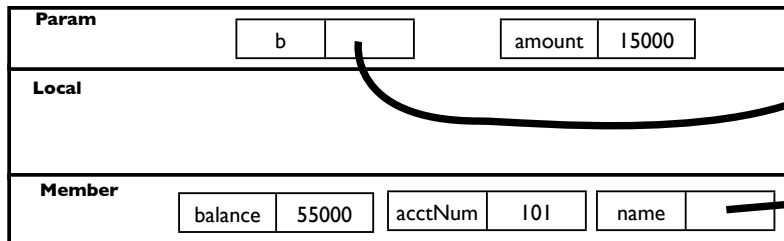
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

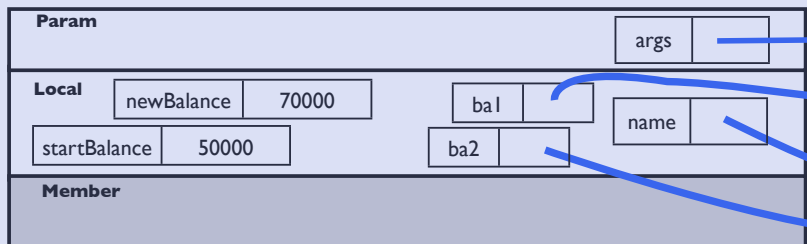
    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

Stack

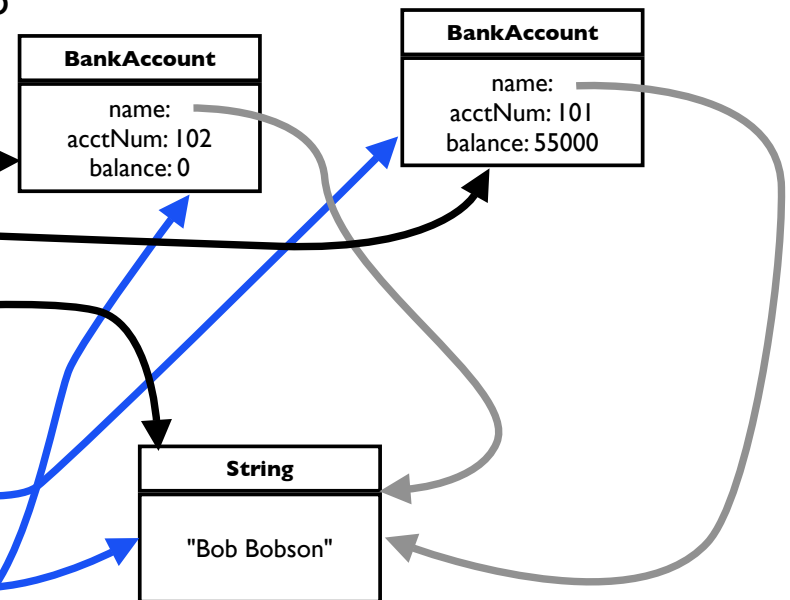
transferMoney



main: 6



Heap



Memory Diagram Walkthrough: 41

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }

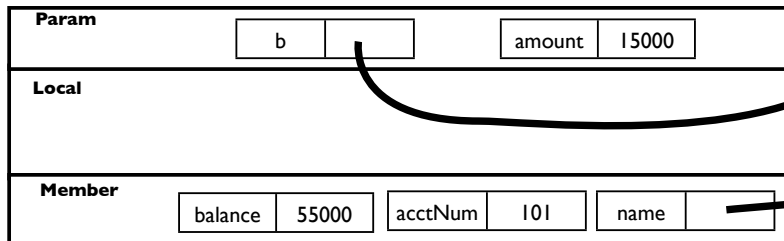
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

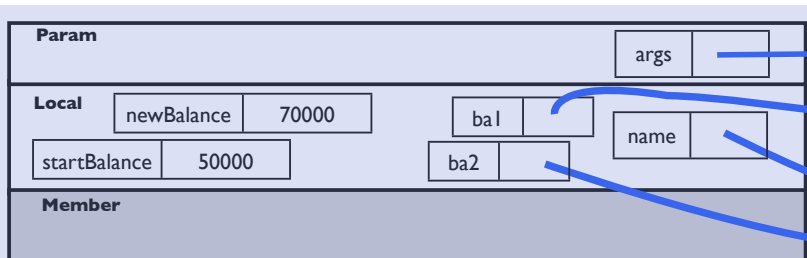
    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

Stack

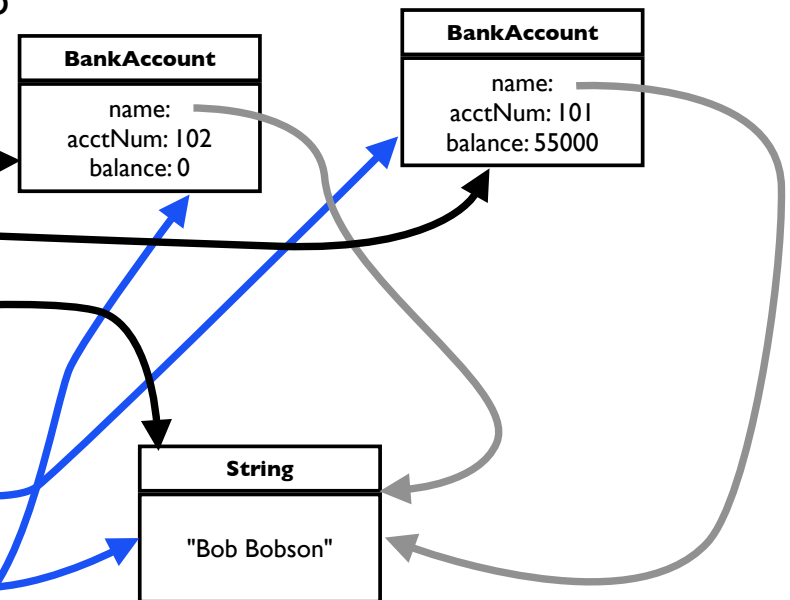
transferMoney



main: 6



Heap



Memory Diagram Walkthrough: 42

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }

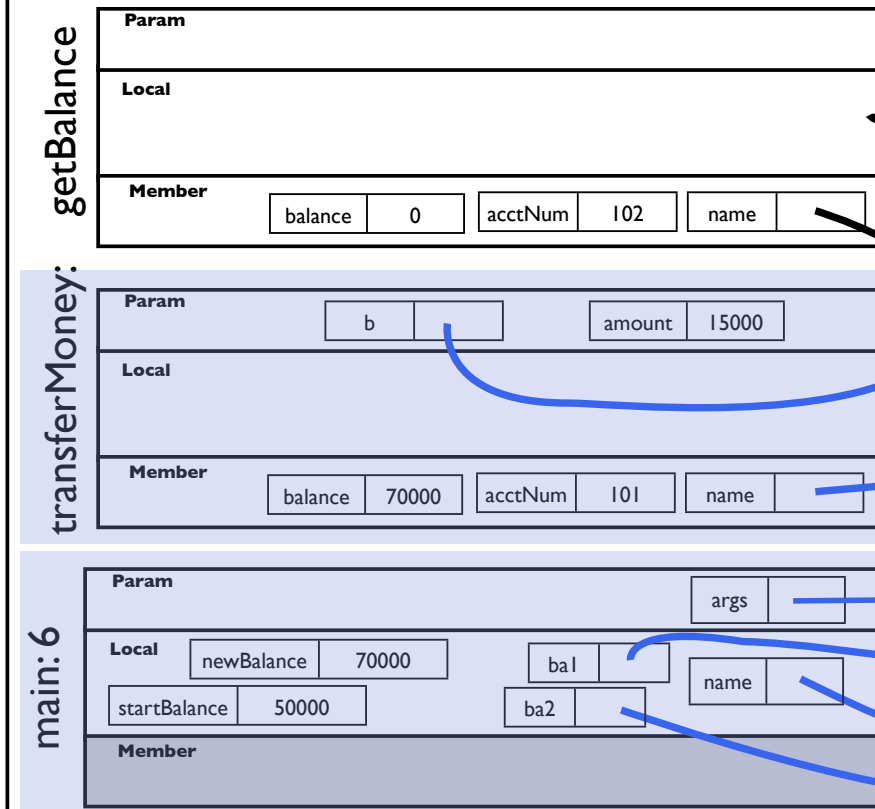
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

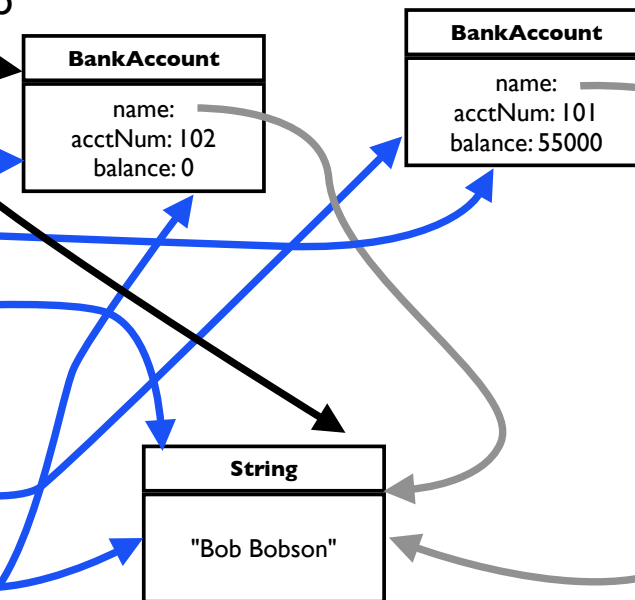
    private void setBalance(int newAmount) { balance = newAmount; }

    public int getBalance() { return balance; }
}
```

Stack



Heap



Memory Diagram Walkthrough: 43

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }

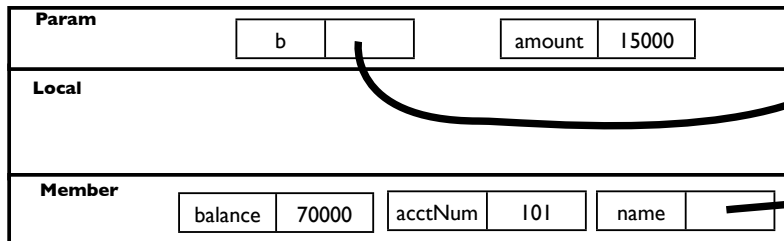
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

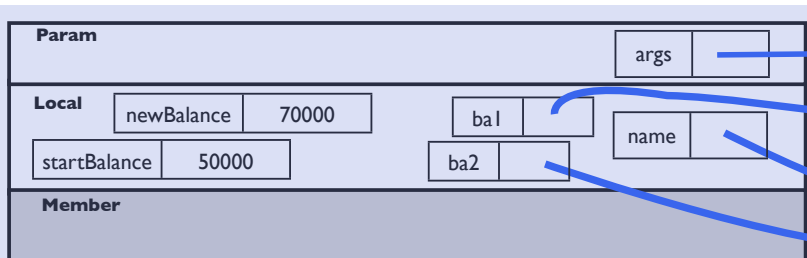
    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

Stack

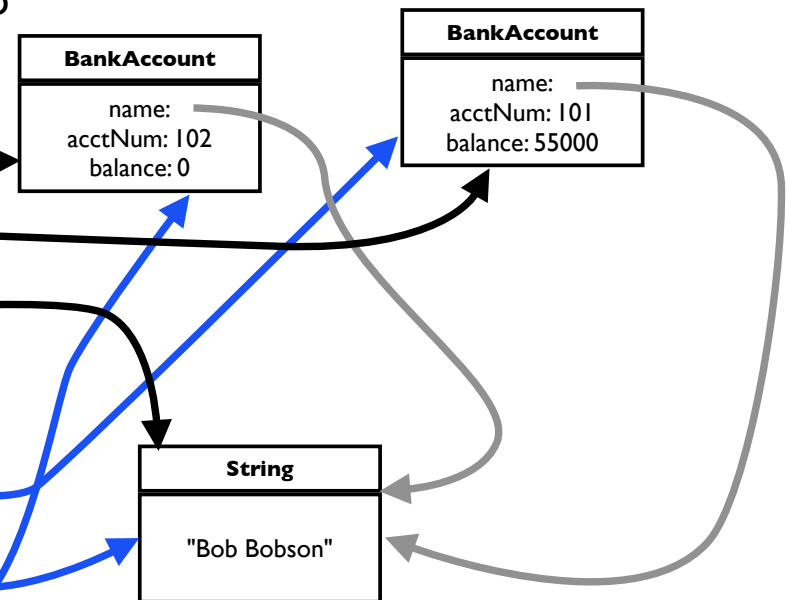
transferMoney



main: 6



Heap



Memory Diagram Walkthrough: 44

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }

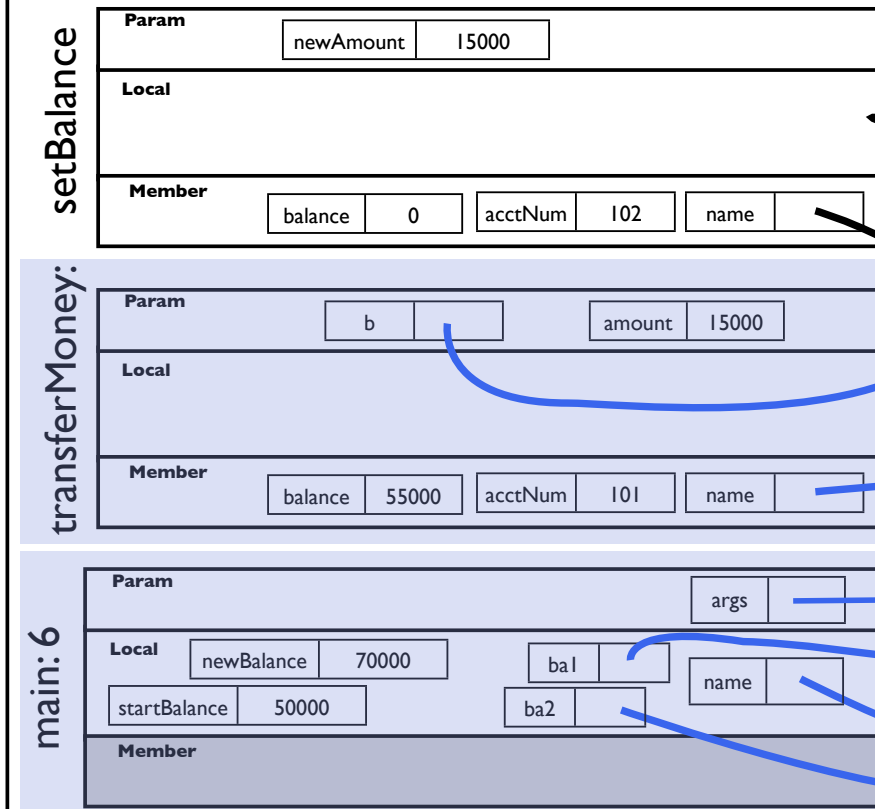
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

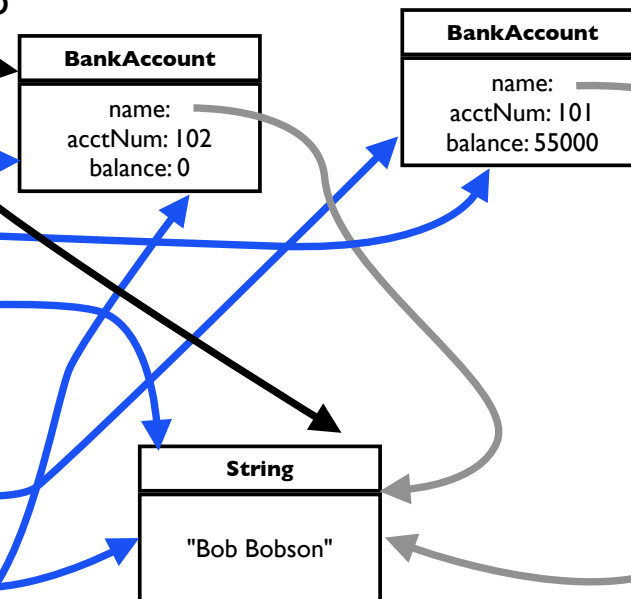
    private void setBalance(int newAmount) { balance = newAmount; }

    public int getBalance() { return balance; }
}
```

Stack



Heap



Memory Diagram Walkthrough: 45

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }

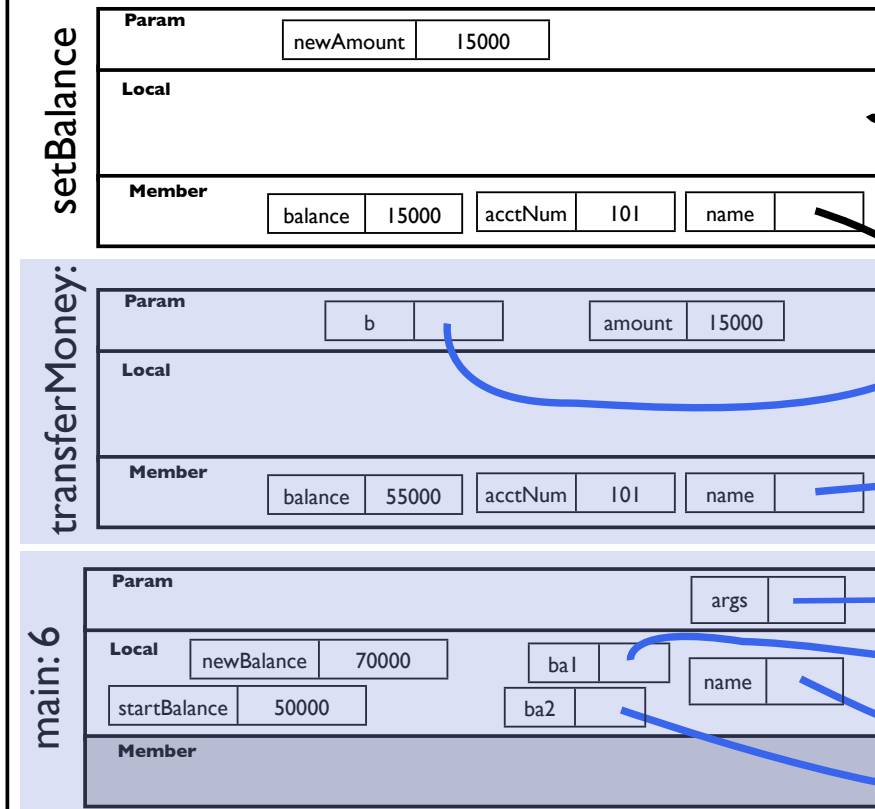
    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }

    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

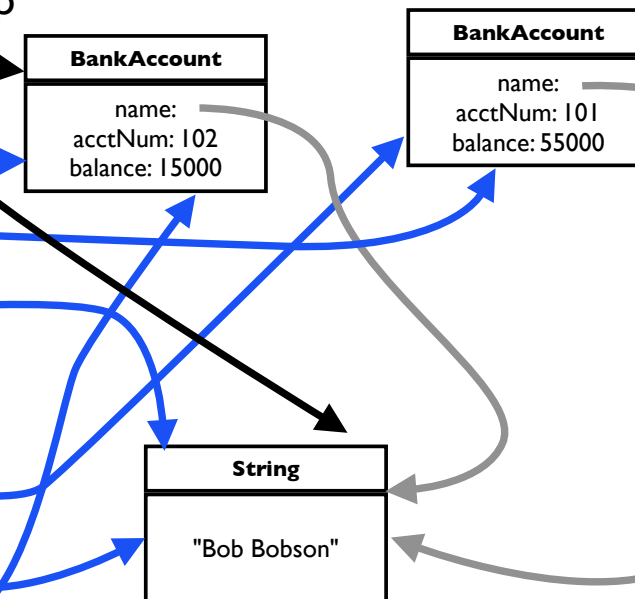
    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

Stack



Heap



Memory Diagram Walkthrough: 46

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name; this.acctNum = number; this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name; this.acctNum = number; this.balance = balance;
    }

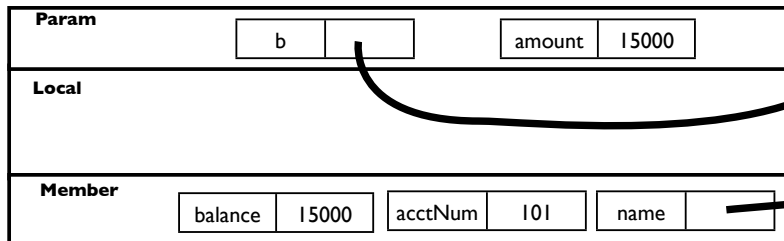
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

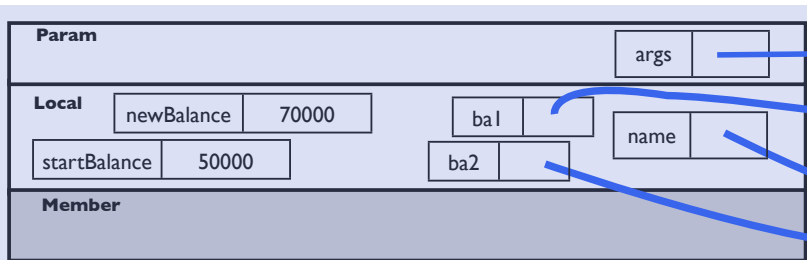
    private void setBalance(int newAmount) { balance = newAmount; }
    public int getBalance() { return balance; }
}
```

Stack

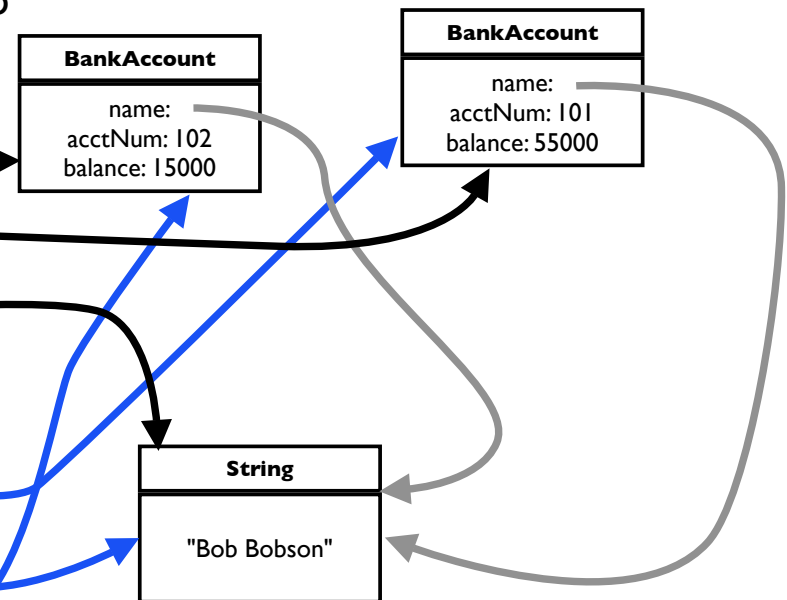
transferMoney



main: 6



Heap



Memory Diagram Walkthrough: 47

BankApplication.java

```
public class BankApplication {
    public static void main(String[] args) {
        String name = "Bob Bobson";
        int startBalance = 50000;
        BankAccount ba1 = new BankAccount(name, 101, startBalance);
        int newBalance = ba1.makeDeposit(20000);
        BankAccount ba2 = new BankAccount(name, 102);
        ba1.transferMoney(ba2, 15000);
    }
}
```

BankAccount.java

```
public class BankAccount {
    public BankAccount(String name, int number) {
        this.name = name;    this.acctNum = number;    this.balance = 0;
    }

    public BankAccount(String name, int number, int balance) {
        this.name = name;    this.acctNum = number;    this.balance = balance;
    }

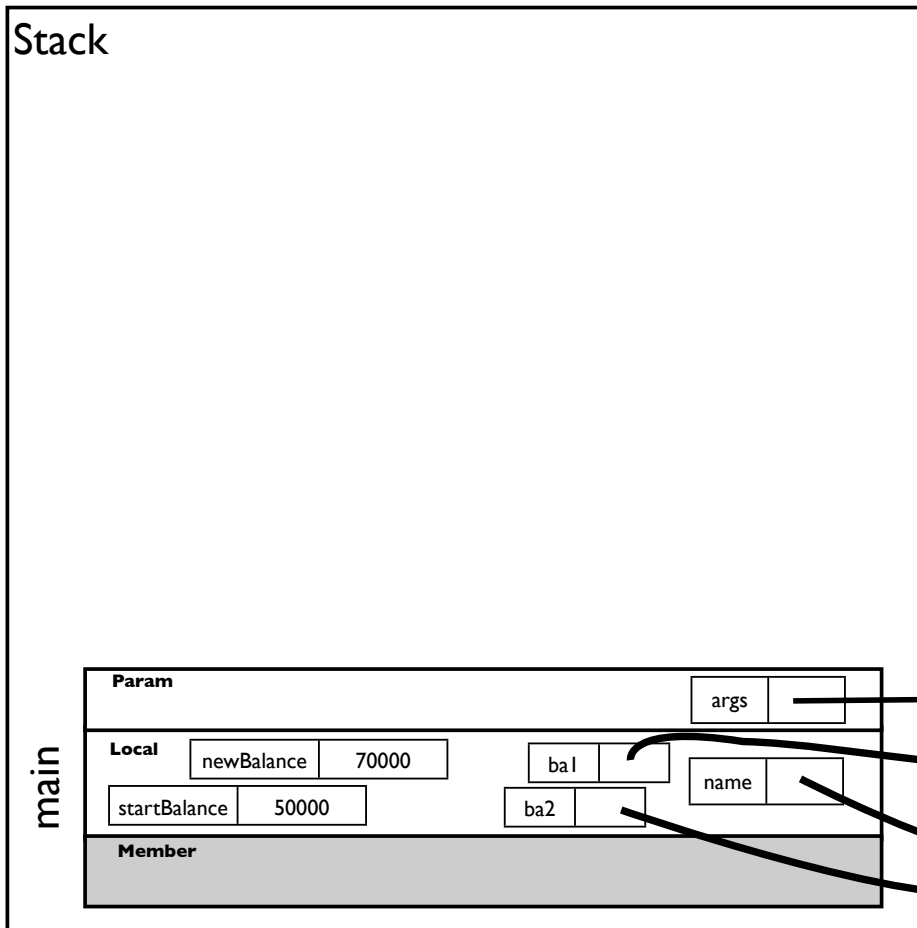
    public int makeDeposit(int amount) {
        int newBalance = this.getBalance() + amount;
        this.setBalance(newBalance);
        return newBalance;
    }

    public void transferMoney(BankAccount b, int amount) {
        this.setBalance(this.getBalance() - amount);
        b.setBalance(b.getBalance() + amount);
    }

    private void setBalance(int newAmount) {    balance = newAmount;    }

    public int getBalance() {    return balance;    }
}
```

Stack



Heap

