INHERITANCE EXAMPLE – ACCOUNT SAVING

Below is the example of inheritance implementation on Java coding and I used Netbean IDE to execute the code. In this example, Account.java is the parent class where it will store the initial amount and account name.

```java
/**
 *
 * @author Najihah Zanudin
 */

class Account
{
    private String name;

    private long amount;

    Account(String name, long amount)
    {
        this.name = name;
        setAmount(amount);
    }

    void deposit(long amount)
    {
        this.amount += amount;
    }

    String getName()
    {
        return name;
    }

    long getAmount()
    {
        return amount;
    }

    void setAmount(long amount)
    {
        this.amount = amount;
    }
}
```

*Figure 1: Account class*

Next, we will show how SavingsAccount and CheckingAccount classes inherit the methods from Account class. Both SavingsAccount and CheckingAccount classes extend the Account class by using java extends as keyword.

We don't have to declare the same thing inside the child classes.

```
/*
 * To change this license header, choose License Headers in Project Properties
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/**
 *
 * @author Najihah Zanudin
 */
class SavingsAccount extends Account
{
    SavingsAccount(long amount)
    {
        super("savings", amount);
    }
}
```

*Figure 2: SavingsAccount class*

```
/*
 * To change this license header, choose License Headers in Project Properties
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/**
 *
 * @author Najihah Zanudin
 */

class CheckingAccount extends Account
{
    CheckingAccount(long amount)
    {
        super("checking", amount);
    }

    void withdraw(long amount)
    {
        setAmount(getAmount() - amount);
    }
}
```

*Figure 3: CheckingAccount class*

Figure below show main() method to call all the method and attributes then compile it to display the output.

```java
/*
 * To change this license header, choose License Headers in Project Properties
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/**
 *
 * @author Najihah Zanudin
 */
public class main {

    public static void main(String[] args)
    {
        SavingsAccount sa = new SavingsAccount(10000);
        System.out.println("account name: " + sa.getName());
        System.out.println("initial amount: " + sa.getAmount());
        sa.deposit(5000);
        System.out.println("new amount after deposit: " + sa.getAmount());

        CheckingAccount ca = new CheckingAccount(20000);
        System.out.println("account name: " + ca.getName());
        System.out.println("initial amount: " + ca.getAmount());
        ca.deposit(6000);
        System.out.println("new amount after deposit: " + ca.getAmount());
        ca.withdraw(3000);
        System.out.println("new amount after withdrawal: " + ca.getAmount());
    }

}
```

*Figure 4: main() method*

This is the output for the inheritance coding shown as above;

```
run:
account name: savings
initial amount: 10000
new amount after deposit: 15000
account name: checking
initial amount: 20000
new amount after deposit: 26000
new amount after withdrawal: 23000
BUILD SUCCESSFUL (total time: 1 second)
```

*Figure 5: output*