27-a

```
class BankAccount {
    private int accountNumber;
    protected double balance;
    public BankAccount(int accountNumber, double balance) {
        this.accountNumber = accountNumber;
        this.balance = balance;
    }
    public int getAccountNumber() {
        return accountNumber;
    }
    public double getBalance() {
        return balance;
    }
    public void deposit(double amount) {
        balance += amount;
        System.out.println("Deposited: $" + amount);
    }
    public void withdraw(double amount) {
        if (amount <= balance) {
            balance -= amount;
            System.out.println("Withdrawn: $" + amount);
        } else {
            System.out.println("Insufficient funds!");
        }
    }
}
class SavingsAccount extends BankAccount {
    private double interestRate;

    public SavingsAccount(int accountNumber, double balance, double interestRate) {
```

```
        super(accountNumber, balance);
        this.interestRate = interestRate;
    }
    public double getInterestRate() {
        return interestRate;
    }
    public void addInterest() {
        double interest = balance * interestRate;
        balance += interest;
        System.out.println("Interest added: $" + interest);
    }
}
class CheckingAccount extends BankAccount {
    private double overdraftLimit;
    public CheckingAccount(int accountNumber, double balance, double overdraftLimit) {
        super(accountNumber, balance);
        this.overdraftLimit = overdraftLimit;
    }
    public double getOverdraftLimit() {
        return overdraftLimit;
    }
    public void withdraw(double amount) {
        if (amount <= balance + overdraftLimit) {
            balance -= amount;
            System.out.println("Withdrawn: $" + amount);
        } else {
            System.out.println("Exceeds overdraft limit!");
        }
    }
}
public class Main {
```

```
    public static void main(String[] args) {
        SavingsAccount savingsAccount = new SavingsAccount(123456, 1000.0, 0.05);
        System.out.println("Savings Account Balance: $" + savingsAccount.getBalance());
        savingsAccount.deposit(500.0);
        savingsAccount.addInterest();
        System.out.println("Savings Account Balance: $" + savingsAccount.getBalance());
        CheckingAccount checkingAccount = new CheckingAccount(654321, 2000.0, 500.0);
        System.out.println("Checking Account Balance: $" + checkingAccount.getBalance());
        checkingAccount.withdraw(2500.0);
        System.out.println("Checking Account Balance: $" + checkingAccount.getBalance());
    }
}
```

27-b

```
import com.pi4j.io.gpio.*;
import com.pi4j.io.gpio.event.*;
public class SensorReading {
    public static void main(String[] args) throws InterruptedException {
        final GpioController gpio = GpioFactory.getInstance();
        final GpioPinDigitalInput sensorPin = gpio.provisionDigitalInputPin(RaspiPin.GPIO_04,
PinPullResistance.PULL_DOWN);
        sensorPin.addListener(new GpioPinListenerDigital() {
            @Override
            public void handleGpioPinDigitalStateChangeEvent(GpioPinDigitalStateChangeEvent event) {
                if (event.getState() == PinState.HIGH) {
                    System.out.println("Light intensity: High");
                } else {
                    System.out.println("Light intensity: Low");
                }
```

```
            }
        });
        while (true) {
            Thread.sleep(1000); // Sleep for 1 second
        }
    }
}
```