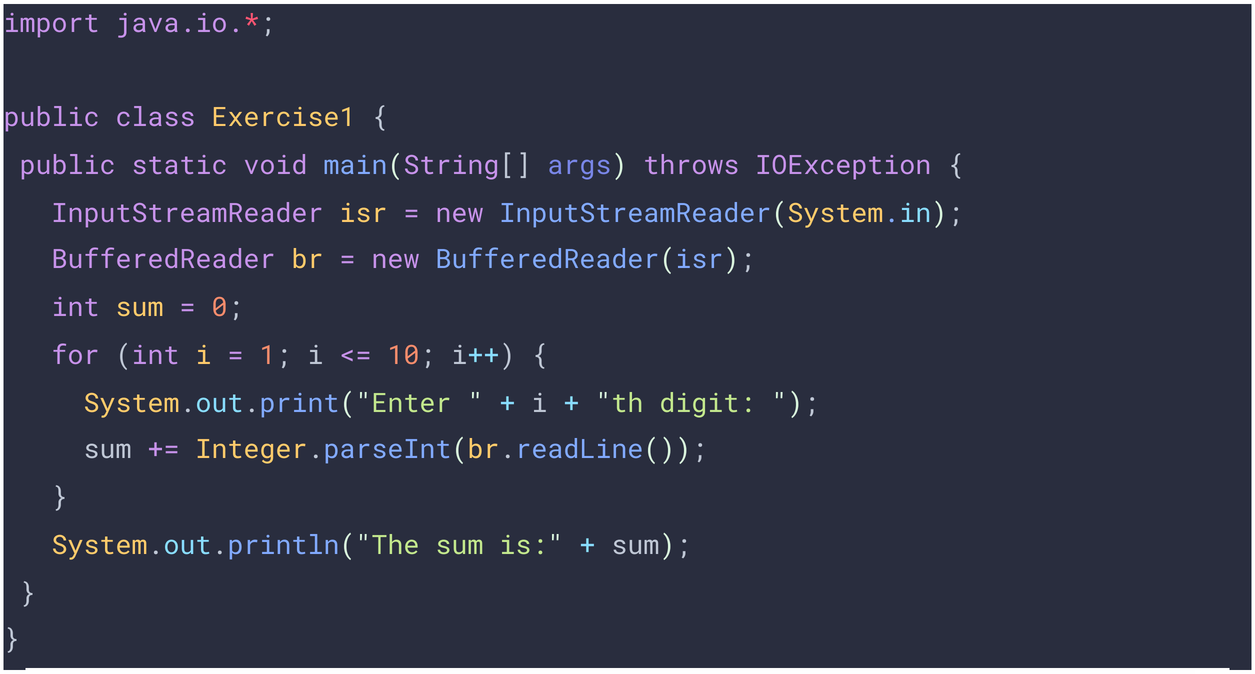
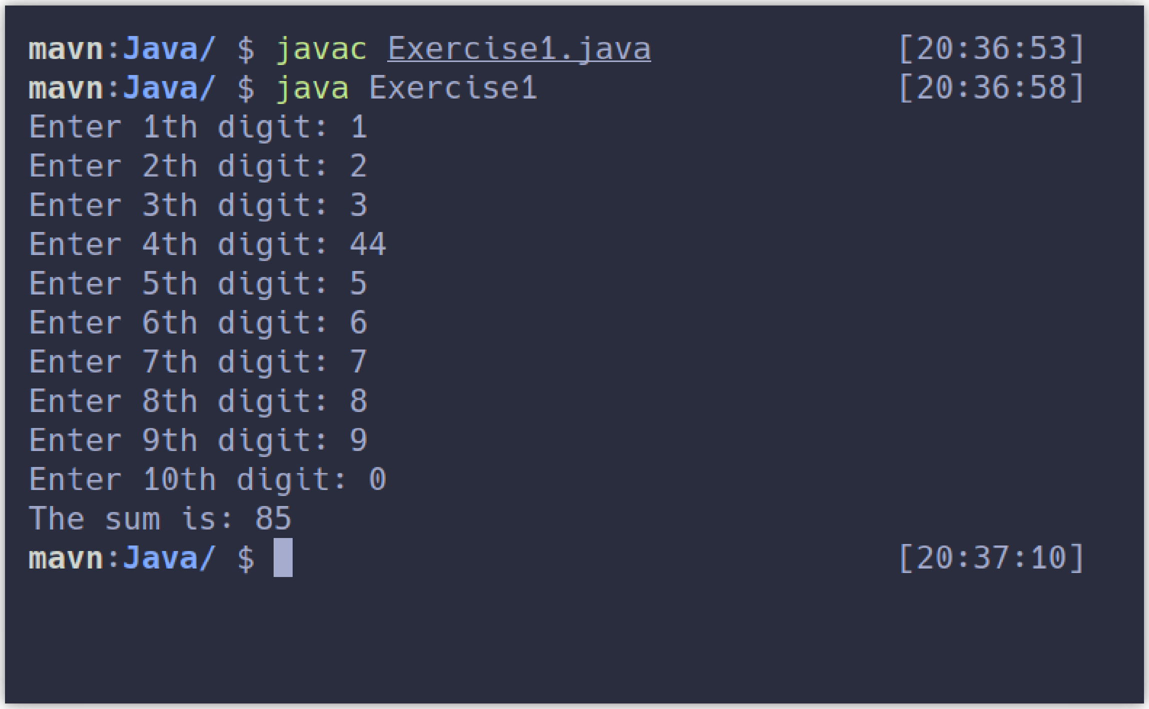
Manavendra Sen | 199302058 | IT 3A

Exercise 1:

Write a program in java to take 10 integer numbers as user input using the BufferedReader and print the sum of these numbers.



```java
import java.io.*;

public class Exercise1 {
 public static void main(String[] args) throws IOException {
   InputStreamReader isr = new InputStreamReader(System.in);
   BufferedReader br = new BufferedReader(isr);
   int sum = 0;
   for (int i = 1; i <= 10; i++) {
     System.out.print("Enter " + i + "th digit: ");
     sum += Integer.parseInt(br.readLine());
   }
   System.out.println("The sum is:" + sum);
 }
}
```

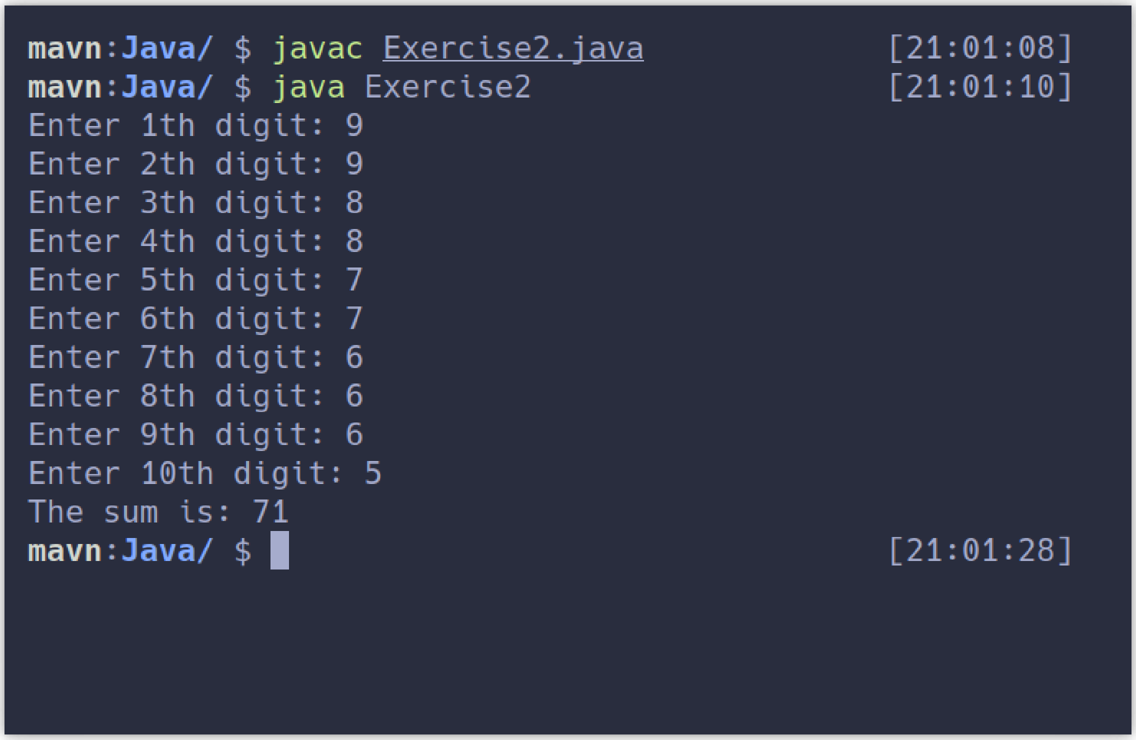


```
mavn:Java/ $ javac Exercise1.java                    [20:36:53]
mavn:Java/ $ java Exercise1                          [20:36:58]
Enter 1th digit: 1
Enter 2th digit: 2
Enter 3th digit: 3
Enter 4th digit: 44
Enter 5th digit: 5
Enter 6th digit: 6
Enter 7th digit: 7
Enter 8th digit: 8
Enter 9th digit: 9
Enter 10th digit: 0
The sum is: 85
mavn:Java/ $                                         [20:37:10]
```

Exercise 2:

Write the program description given in Exercise1 in java using the Scanner class [name the file as Exercise2.java]

```java
import java.util.Scanner;

public class Exercise2 {
 public static void main(String[] args) {
   Scanner sc = new Scanner(System.in);
   int sum = 0;
   for (int i = 1; i <= 10; i++) {
     System.out.print("Enter " + i + "th digit: ");
     sum += sc.nextInt();
   }
   System.out.println("The sum is: " + sum);
 }
}
```



```
mavn:Java/ $ javac Exercise2.java                    [21:01:08]
mavn:Java/ $ java Exercise2                          [21:01:10]
Enter 1th digit: 9
Enter 2th digit: 9
Enter 3th digit: 8
Enter 4th digit: 8
Enter 5th digit: 7
Enter 6th digit: 7
Enter 7th digit: 6
Enter 8th digit: 6
Enter 9th digit: 6
Enter 10th digit: 5
The sum is: 71
mavn:Java/ $                                         [21:01:28]
```

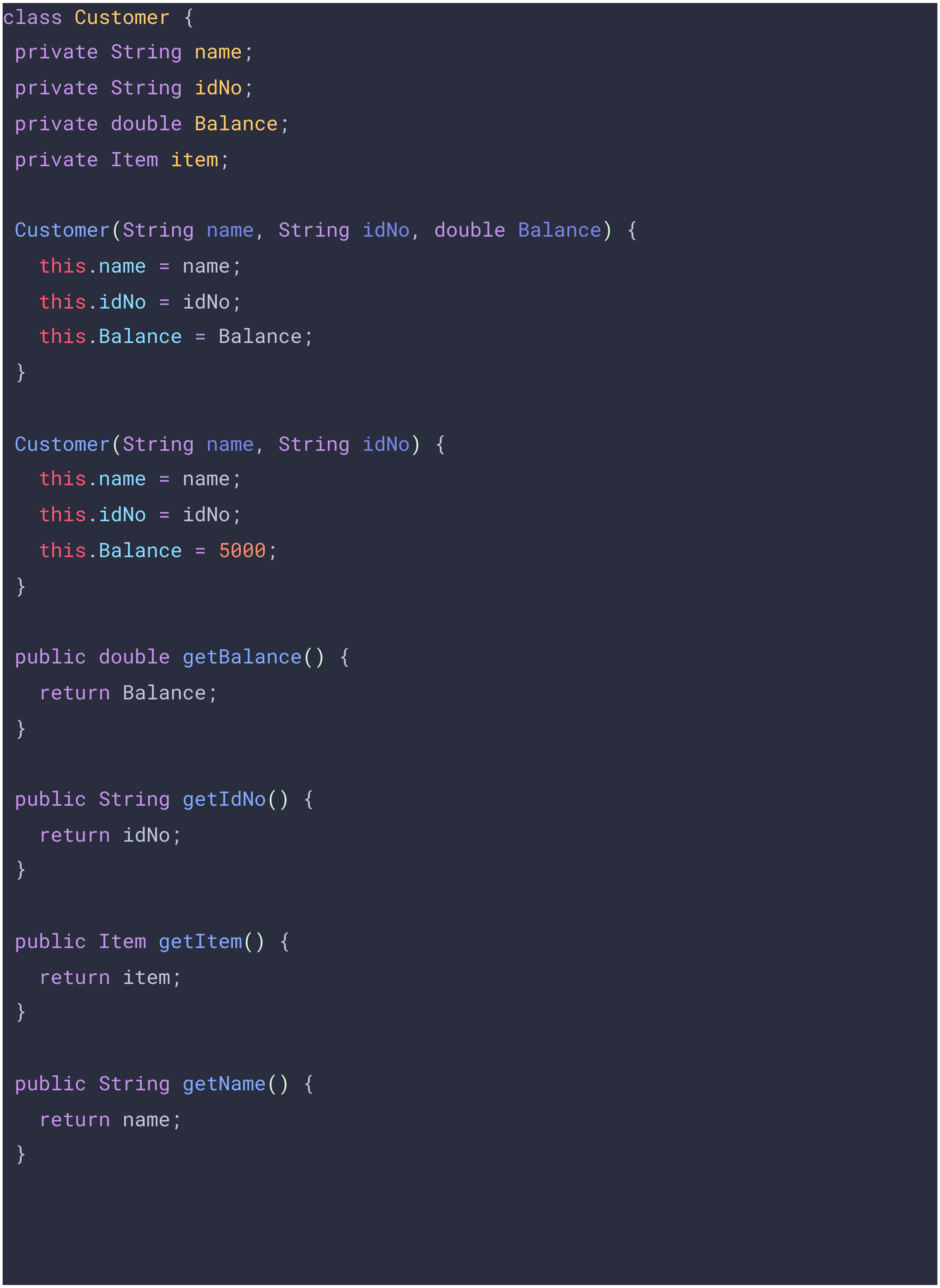
2.1 Write java implementation for a class named '*Item*' which encapsulates the details of items to be purchased by the customer of the XYZ shop.

2.2 Write the java implementation for a class named 'Customer' which encapsulates the details of registered customers of the XYZ shop who buy *Items* (class is described above in 2.1) online.

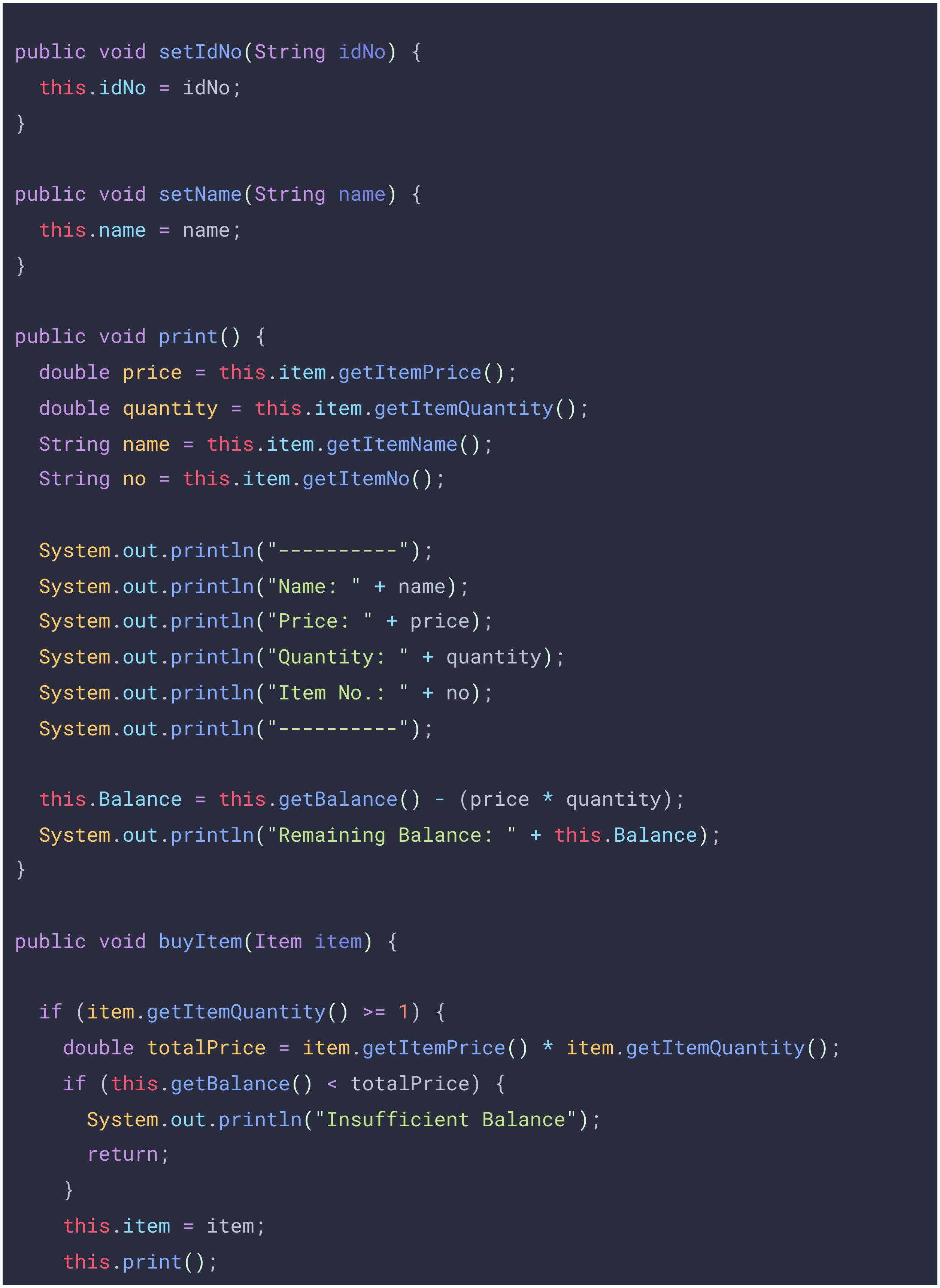
2.3 Write a Test class named TestStore.java

```java
package Practicals;

public class Item {
 private String ItemName;
 private String ItemNo;
 private int ItemQuantity;
 private double ItemPrice;

 // Constructor
 Item(String ItemName, String ItemNo, int ItemQuantity, double ItemPrice) {
   this.ItemName = ItemName;
   this.ItemNo = ItemNo;
   this.ItemQuantity = ItemQuantity;
   this.ItemPrice = ItemPrice;
 }

 Item(String ItemName, String ItemNo, int ItemQuantity) {
   this.ItemName = ItemName;
   this.ItemNo = ItemNo;
   this.ItemQuantity = ItemQuantity;
   this.ItemPrice = 500;
 }

 Item(String ItemName, String ItemNo) {
   this.ItemName = ItemName;
   this.ItemNo = ItemNo;
   this.ItemQuantity = 1;
   this.ItemPrice = 500;
```

```java
  }

  public String getItemName() {
    return ItemName;
  }

  public String getItemNo() {
    return ItemNo;
  }

  public double getItemPrice() {
    return ItemPrice;
  }

  public int getItemQuantity() {
    return ItemQuantity;
  }

  public void setItemName(String itemName) {
    ItemName = itemName;
  }

  public void setItemNo(String itemNo) {
    ItemNo = itemNo;
  }

  public void setItemPrice(double itemPrice) {
    ItemPrice = itemPrice;
  }

  public void setItemQuantity(int itemQuantity) {
    ItemQuantity = itemQuantity;
  }
}
```



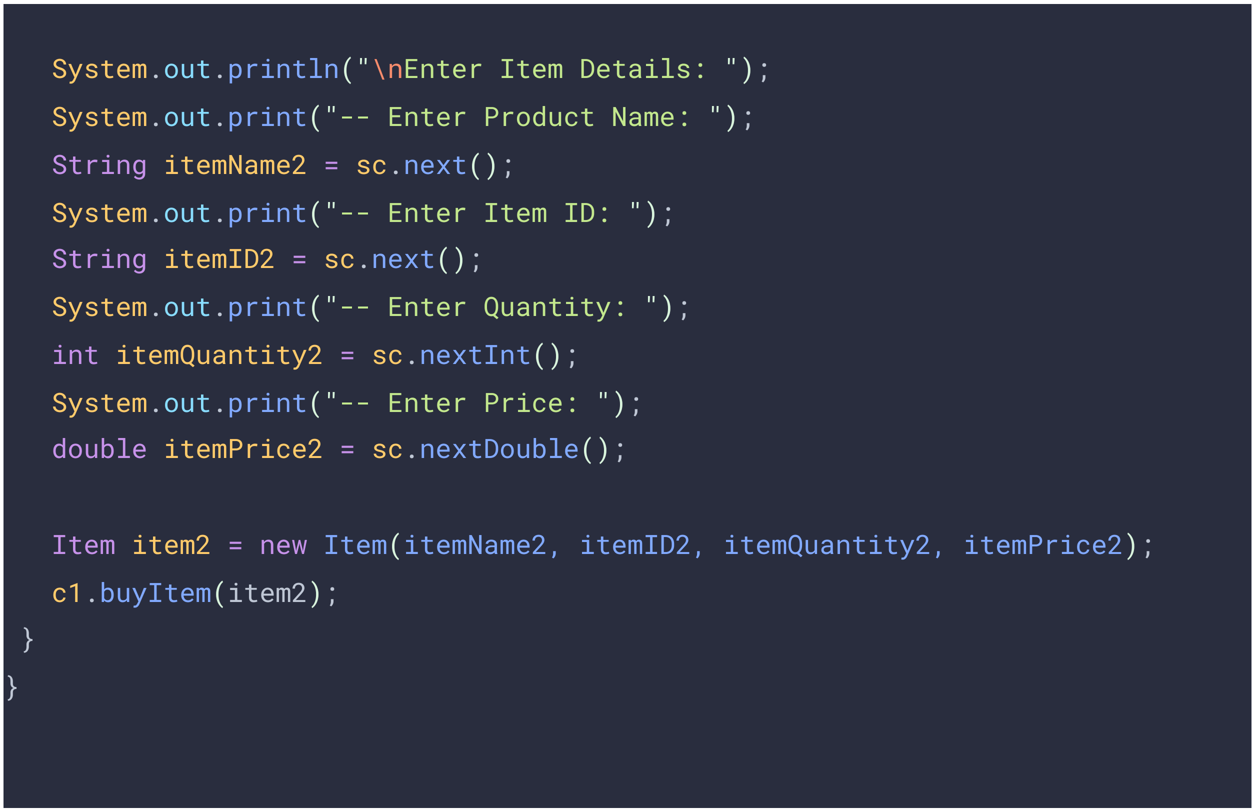
```java
class Customer {
 private String name;
 private String idNo;
 private double Balance;
 private Item item;

 Customer(String name, String idNo, double Balance) {
   this.name = name;
   this.idNo = idNo;
   this.Balance = Balance;
 }

 Customer(String name, String idNo) {
   this.name = name;
   this.idNo = idNo;
   this.Balance = 5000;
 }

 public double getBalance() {
   return Balance;
 }

 public String getIdNo() {
   return idNo;
 }

 public Item getItem() {
   return item;
 }

 public String getName() {
   return name;
 }
```



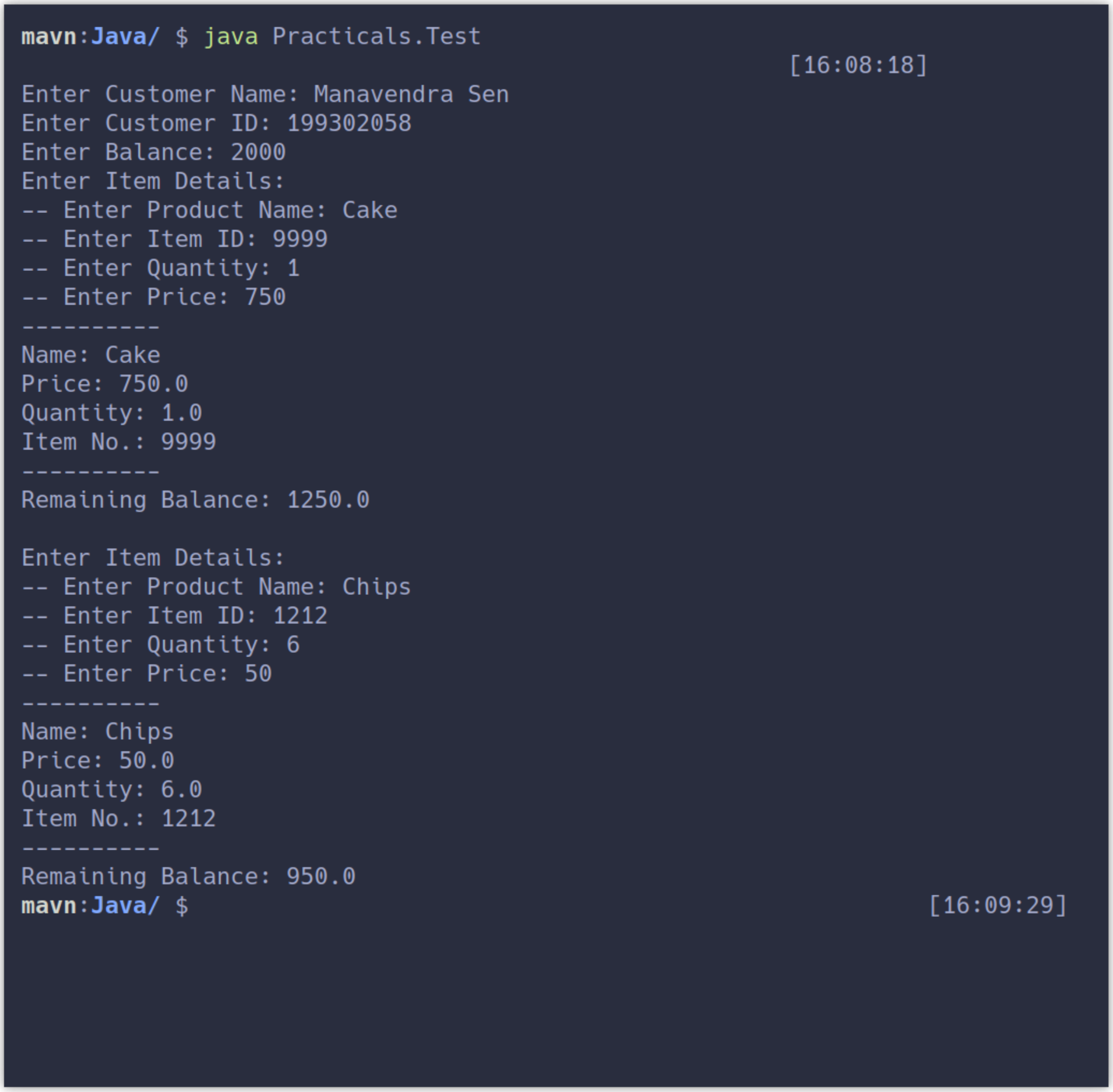
```java
public void setIdNo(String idNo) {
  this.idNo = idNo;
}

public void setName(String name) {
  this.name = name;
}

public void print() {
  double price = this.item.getItemPrice();
  double quantity = this.item.getItemQuantity();
  String name = this.item.getItemName();
  String no = this.item.getItemNo();

  System.out.println("----------");
  System.out.println("Name: " + name);
  System.out.println("Price: " + price);
  System.out.println("Quantity: " + quantity);
  System.out.println("Item No.: " + no);
  System.out.println("----------");

  this.Balance = this.getBalance() - (price * quantity);
  System.out.println("Remaining Balance: " + this.Balance);
}

public void buyItem(Item item) {

  if (item.getItemQuantity() >= 1) {
    double totalPrice = item.getItemPrice() * item.getItemQuantity();
    if (this.getBalance() < totalPrice) {
      System.out.println("Insufficient Balance");
      return;
    }
    this.item = item;
    this.print();
```

```
   } else
     System.out.println("Order is not valid");
 }
}
```

```
package Practicals;
import java.util.Scanner;

class Test {
 public static void main(String[] args) {
   Scanner sc = new Scanner(System.in);
   System.out.print("Enter Customer Name: ");
   String name = sc.nextLine();
   System.out.print("Enter Customer ID: ");
   String id = sc.nextLine();
   System.out.print("Enter Balance: ");
   double balance = sc.nextDouble();

   Customer c1 = new Customer(name, id, balance);

   // input for item
   System.out.println("Enter Item Details: ");
   System.out.print("-- Enter Product Name: ");
   String itemName = sc.next();
   System.out.print("-- Enter Item ID: ");
   String itemID = sc.next();
   System.out.print("-- Enter Quantity: ");
   int itemQuantity = sc.nextInt();
   System.out.print("-- Enter Price: ");
   double itemPrice = sc.nextDouble();

   Item item1 = new Item(itemName, itemID, itemQuantity, itemPrice);
   c1.buyItem(item1);
```



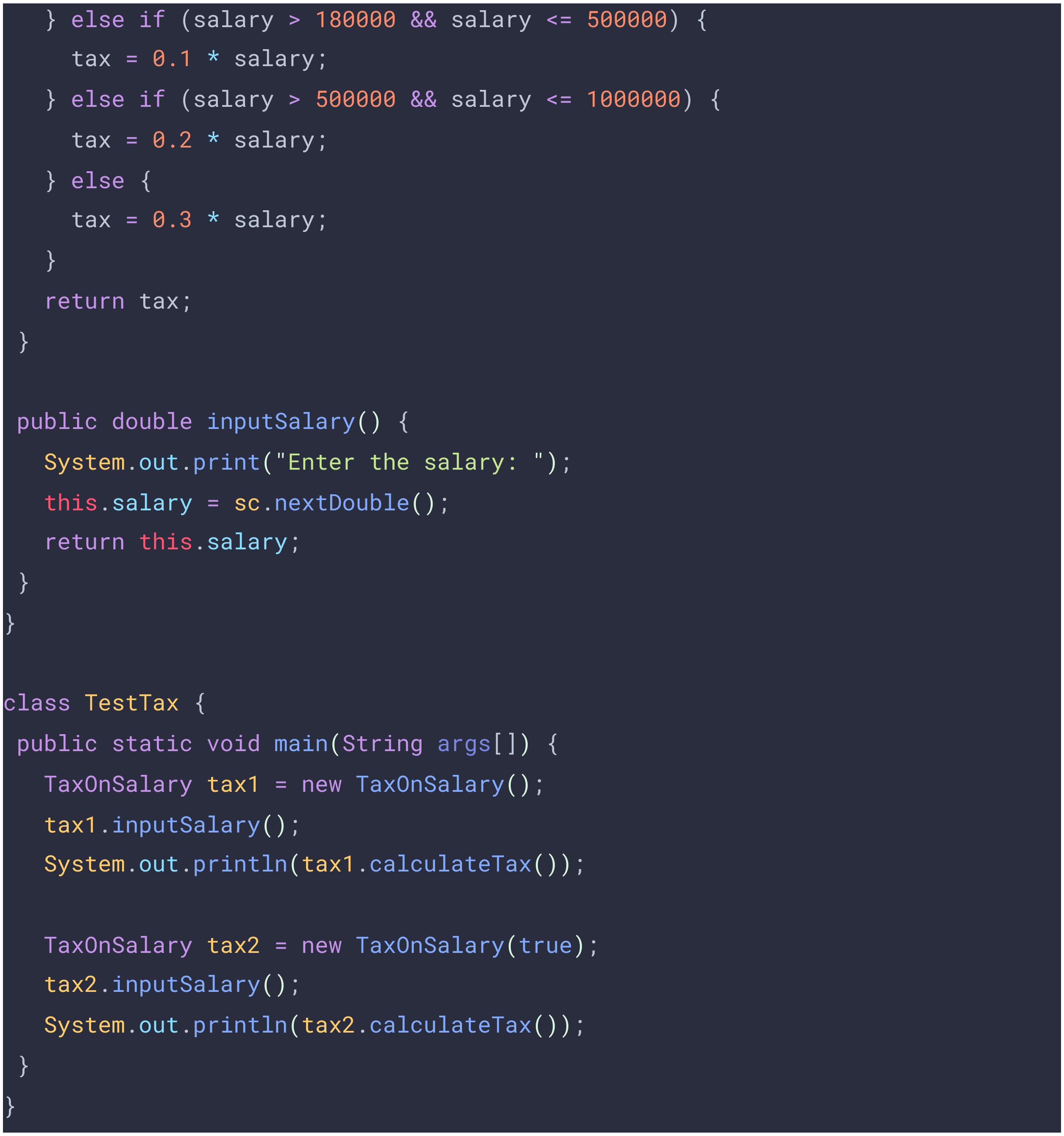
```java
   System.out.println("\nEnter Item Details: ");
   System.out.print("-- Enter Product Name: ");
   String itemName2 = sc.next();
   System.out.print("-- Enter Item ID: ");
   String itemID2 = sc.next();
   System.out.print("-- Enter Quantity: ");
   int itemQuantity2 = sc.nextInt();
   System.out.print("-- Enter Price: ");
   double itemPrice2 = sc.nextDouble();

   Item item2 = new Item(itemName2, itemID2, itemQuantity2, itemPrice2);
   c1.buyItem(item2);
 }
}
```



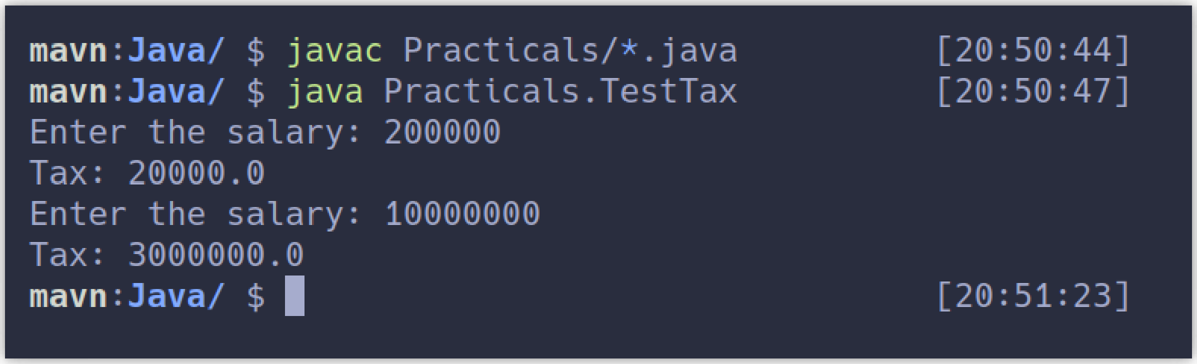
```
mavn:Java/ $ java Practicals.Test
                                                       [16:08:18]
Enter Customer Name: Manavendra Sen
Enter Customer ID: 199302058
Enter Balance: 2000
Enter Item Details:
-- Enter Product Name: Cake
-- Enter Item ID: 9999
-- Enter Quantity: 1
-- Enter Price: 750
----------
Name: Cake
Price: 750.0
Quantity: 1.0
Item No.: 9999
----------
Remaining Balance: 1250.0

Enter Item Details:
-- Enter Product Name: Chips
-- Enter Item ID: 1212
-- Enter Quantity: 6
-- Enter Price: 50
----------
Name: Chips
Price: 50.0
Quantity: 6.0
Item No.: 1212
----------
Remaining Balance: 950.0
mavn:Java/ $                                                     [16:09:29]
```

Exercises 3:
Write the java implementation for a class named ‘*TaxOnSalary*’ to calculate tax on salary.

```
package Practicals;
import java.util.Scanner;

public class TaxOnSalary {
 private double salary;
 private boolean isPANsubmitted;

 Scanner sc = new Scanner(System.in);

 TaxOnSalary() {
   salary = 0.0;
   isPANsubmitted = false;
 }

 TaxOnSalary(boolean pan) {
   salary = 1000.0;
   isPANsubmitted = pan;
 }

 public double getSalary() {
   return salary;
 }

 public boolean getIsPANsubmitted() {
   return isPANsubmitted;
 }

 public double calculateTax() {
   double tax;
   if (salary <= 180000 && isPANsubmitted == true) {
     tax = 0;
   } else if (salary <= 180000 && isPANsubmitted == false) {
     tax = 0.05 * salary;
```



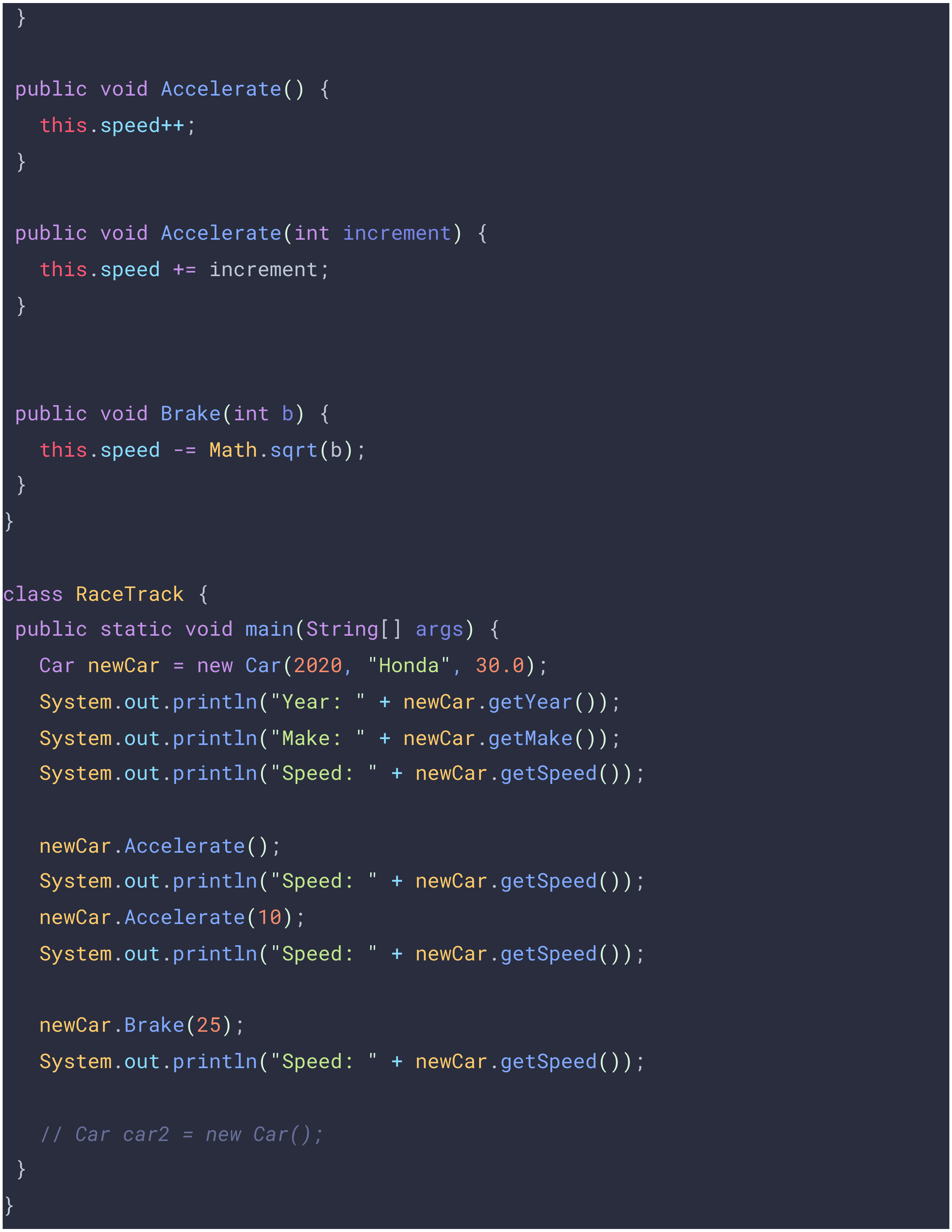
```
   } else if (salary > 180000 && salary <= 500000) {
     tax = 0.1 * salary;
   } else if (salary > 500000 && salary <= 1000000) {
     tax = 0.2 * salary;
   } else {
     tax = 0.3 * salary;
   }
   return tax;
 }

 public double inputSalary() {
   System.out.print("Enter the salary: ");
   this.salary = sc.nextDouble();
   return this.salary;
 }
}

class TestTax {
 public static void main(String args[]) {
   TaxOnSalary tax1 = new TaxOnSalary();
   tax1.inputSalary();
   System.out.println(tax1.calculateTax());

   TaxOnSalary tax2 = new TaxOnSalary(true);
   tax2.inputSalary();
   System.out.println(tax2.calculateTax());
 }
}
```



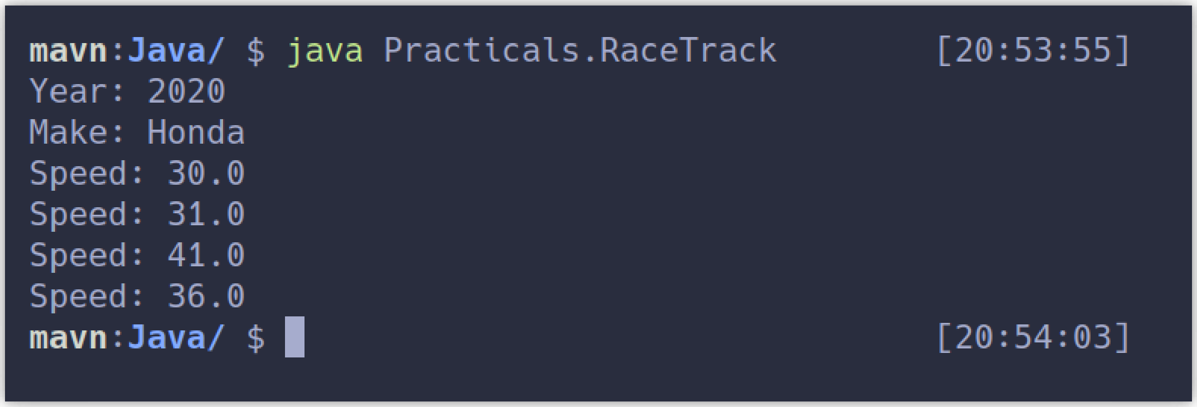
```
mavn:Java/ $ javac Practicals/*.java                      [20:50:44]
mavn:Java/ $ java Practicals.TestTax                      [20:50:47]
Enter the salary: 200000
Tax: 20000.0
Enter the salary: 10000000
Tax: 3000000.0
mavn:Java/ $                                              [20:51:23]
```

Exercise 4:
A. Define a class Car which encapsulates following attributes and methods.
B. Define a RaceTrack class that has main method do the following activities.

```
package Practicals;

public class Car {
 private int year;
 private String make;
 private double speed;

 Car(int year, String make, double speed) {
   this.year = year;
   this.make = make;
   this.speed = speed;
 }

 public String getMake() {
   return make;
 }

 public double getSpeed() {
   return speed;
 }

 public int getYear() {
   return year;
```



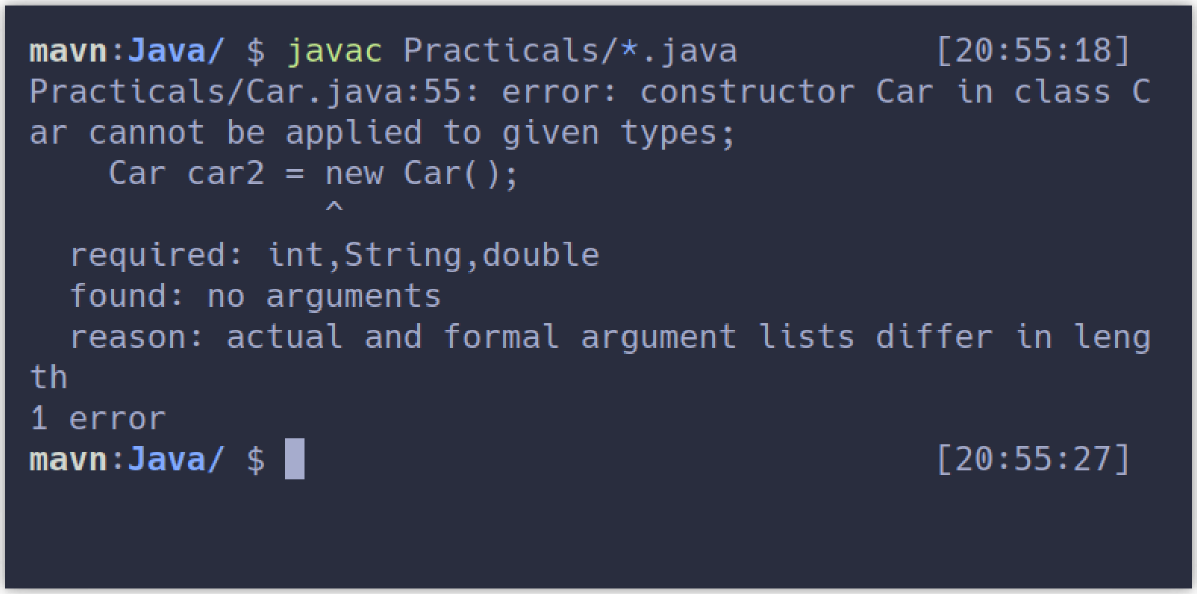
```
 }

 public void Accelerate() {
   this.speed++;
 }

 public void Accelerate(int increment) {
   this.speed += increment;
 }


 public void Brake(int b) {
   this.speed -= Math.sqrt(b);
 }
}

class RaceTrack {
 public static void main(String[] args) {
   Car newCar = new Car(2020, "Honda", 30.0);
   System.out.println("Year: " + newCar.getYear());
   System.out.println("Make: " + newCar.getMake());
   System.out.println("Speed: " + newCar.getSpeed());

   newCar.Accelerate();
   System.out.println("Speed: " + newCar.getSpeed());
   newCar.Accelerate(10);
   System.out.println("Speed: " + newCar.getSpeed());

   newCar.Brake(25);
   System.out.println("Speed: " + newCar.getSpeed());

   // Car car2 = new Car();
 }
}
```



```
mavn:Java/ $ java Practicals.RaceTrack         [20:53:55]
Year: 2020
Make: Honda
Speed: 30.0
Speed: 31.0
Speed: 41.0
Speed: 36.0
mavn:Java/ $                                   [20:54:03]
```

- Now, create a new Car object without passing arguments. Compile and observe the output

```
Car car2 = new Car();
```



```
mavn:Java/ $ javac Practicals/*.java           [20:55:18]
Practicals/Car.java:55: error: constructor Car in class C
ar cannot be applied to given types;
    Car car2 = new Car();
               ^
  required: int,String,double
  found: no arguments
  reason: actual and formal argument lists differ in leng
th
1 error
mavn:Java/ $                                   [20:55:27]
```

Explanation: We have used a parameterized Constructor, so we need to supply the non parameterized constructor also, if we want to use it, it won't be supplied by JRE.