

Basavaraj Aili

KodNest CSR June 2023

Assignment: Programming in JAVA (08/07/2023)

Question: 11 **Problem: The Planet Explorer**

You are a scientist at a space research center. The center is currently researching various planets and their properties. One of the properties they are interested in is the surface area of these planets. Considering that a planet can be approximated as a sphere, your task is to create a method that calculates the surface area of a sphere given its radius.

****Hint:**** The formula to calculate the surface area of a sphere is:

$$A = 4\pi r^2$$

Where:

- A is the surface area of the sphere
- r is the radius of the sphere

****Class Definition:****

```
public class PlanetExplorer {  
    public double calculateSurfaceArea(double radius);  
}
```

****Inputs:****

The method calculateSurfaceArea(double radius) will receive one parameter:

- radius : a double representing the radius of the sphere (planet).

****Outputs:****

The method will return a double - the surface area of the sphere.

****Example:****

****Sample Input:****

```
PlanetExplorer explorer = new PlanetExplorer();
```

```
explorer.calculateSurfaceArea(3.0);
```

****Sample Output:****

113.10

****Note:****

In the sample input, the radius of the planet (sphere) is 3.0 units. The surface area is $4\pi \times 3^2 = 113.10$, so the method returns 113.10 as the output. Your method will be essential in helping the scientists at the space research center in understanding the properties of various planets. Good luck, scientist!

Solution 11:

```
public class PlanetExplorer {  
    public double calculateSurfaceArea(double radius) {  
        return 4*Math.PI*Math.pow(radius,2);  
    }  
}  
  
import java.util.Scanner;  
  
public class PlanetExplorerApp {  
    public static void main(String[] args) {  
        Scanner scan=new Scanner(System.in);  
        System.out.println("Enter the radius: ");  
        double radius=scan.nextDouble();  
        PlanetExplorer explorer = new PlanetExplorer();  
        System.out.println(explorer.calculateSurfaceArea(radius));  
        scan.close();  
    }  
}
```

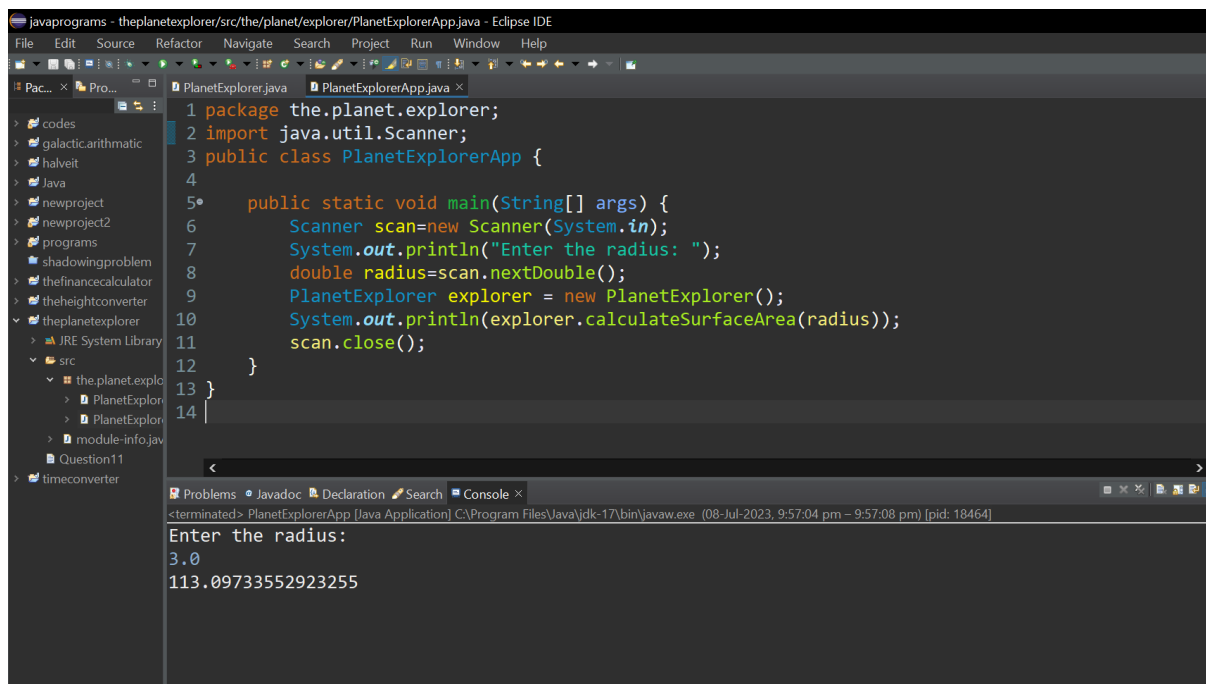
Output:

Enter the radius:

3.0

113.09733552923255

Snapshot 11



The screenshot shows the Eclipse IDE with the file `PlanetExplorerApp.java` open. The code defines a `PlanetExplorerApp` class with a `main` method that uses a `Scanner` to read a radius and calls `calculateSurfaceArea` on a `PlanetExplorer` object. The console output shows the program running and displaying the radius 3.0 and the calculated surface area 113.09733552923255.

```
1 package the.planet.explorer;
2 import java.util.Scanner;
3 public class PlanetExplorerApp {
4
5     public static void main(String[] args) {
6         Scanner scan=new Scanner(System.in);
7         System.out.println("Enter the radius: ");
8         double radius=scan.nextDouble();
9         PlanetExplorer explorer = new PlanetExplorer();
10        System.out.println(explorer.calculateSurfaceArea(radius));
11        scan.close();
12    }
13 }
14
```

Console Output:

```
<terminated> PlanetExplorerApp [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (08-Jul-2023, 9:57:04 pm - 9:57:08 pm) [pid: 18464]
Enter the radius:
3.0
113.09733552923255
```

Question: 12 ****Problem: The Height Converter****

You are part of a sports data management team. The team is developing a new feature for their application where the heights of players, currently recorded in inches, need to be displayed in feet for an international audience. Your task is to create a method that takes a height given in inches and converts it into feet.

****Hint:**** The conversion factor from inches to feet is 1 foot = 12 inches.

****Class Definition:****

```
public class HeightConverter {

    public double convertInchesToFeet(double inches);

}
```

****Inputs:****

The method `convertInchesToFeet(double inches)` will receive one parameter:

- inches ($0 \leq \text{inches} \leq 10^9$): a double representing the height in inches.

****Outputs:****

The method will return a double - the height converted to feet.

****Example:********Sample Input:****

```
HeightConverter converter = new HeightConverter();
```

```
converter.convertInchesToFeet(72.0);
```

****Sample Output:****

6.00

****Note:****

In the sample input, the height of the player is 72.0 inches. The height in feet is $72/12 = 6.00$, so the method returns 6.00 as the output. Your method will be crucial in helping the sports data management team present the data in a format familiar to the international audience. Good luck, data manager!

Solution 12:

```
public class HeightConverter {  
    public double convertInchesToFeet(double inches) {  
        return inches/12;  
    }  
}  
  
import java.util.Scanner;  
  
public class HeightConverterApp {  
    public static void main(String[] args) {  
        Scanner scan=new Scanner(System.in);  
        System.out.println("Enter the inches as Real Number: ");
```

```

        double inches=scan.nextDouble();

        HeightConverter converter = new HeightConverter();

        System.out.println(converter.convertInchesToFeet(inches));

        scan.close();

    }

}

```

Output:

Enter the inches as Real Number:

72.0

6.0

Snapshot 12

```

1 package theheightconverter;
2 import java.util.Scanner;
3 public class HeightConverterApp {
4
5     public static void main(String[] args) {
6         Scanner scan=new Scanner(System.in);
7         System.out.println("Enter the inches as Real Number: ");
8         double inches=scan.nextDouble();
9         HeightConverter converter = new HeightConverter();
10        System.out.println(converter.convertInchesToFeet(inches));
11        scan.close();
12    }
13 }
14

```

Problems • Javadoc • Declaration • Search • Console ×

<terminated> HeightConverterApp [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (08-Jul-2023, 10:03:12 pm – 10:03:26 pm) [pid: 20084]

Enter the inches as Real Number:
72.0
6.0

Question: 13 ****Problem: The Finance Calculator****

You are a software developer at a financial technology company. The company is building a new feature in their app that calculates the simple interest for users wanting to take out loans. Your task is to create a method that calculates the simple interest given the principal amount, rate of interest, and time.

****Hint:**** The formula to calculate simple interest is:

$$I = P * R * T$$

Where:

- I is the simple interest
- P is the principal amount
- R is the rate of interest (in decimal)
- T is the time (in years)

****Class Definition:****

```
public class FinanceCalculator {  
  
    public double calculateSimpleInterest(double principal, double rate, double time);  
  
}
```

****Inputs:****

The method calculateSimpleInterest(double principal, double rate, double time) will receive three parameters:

- principal ($1 \leq \text{principal} \leq 10^9$): a double representing the principal amount.
- rate ($0 \leq \text{rate} \leq 1$): a double representing the rate of interest in decimal.
- time ($1 \leq \text{time} \leq 10^5$): a double representing the time in years.

****Outputs:****

The method will return a double - the simple interest.

****Example:****

****Sample Input:****

```
FinanceCalculator calculator = new FinanceCalculator();
```

```
calculator.calculateSimpleInterest(1000.0, 0.05, 2.0);
```

****Sample Output:****

100.00

****Note:****

In the sample input, the principal amount is 1000.0 units, the rate of interest is 0.05, and the time is 2.0 years. The simple interest is $1000.0 * 0.05 * 2.0 = 100.00$, so the method returns 100.00 as the output. Your method will be crucial in helping users plan their finances. Good luck, developer!

Solution 13:

```
public class FinanceCalculator {  
  
    public double calculateSimpleInterest(double principal, double rate, double  
time) {  
  
        return principal*rate*time;  
  
    }  
  
}
```

```
import java.util.Scanner;
```

```
public class FinanceCalculatoApp {  
  
    public static void main(String[] args) {  
  
        Scanner scan=new Scanner(System.in);  
  
        System.out.println("Enter the Principal as Real number");  
  
        double principal=scan.nextDouble();  
  
        System.out.println("Enter the Rate of Interest as Real number");  
  
        double rate=scan.nextDouble();  
  
        System.out.println("Enter the Time as Real number");  
  
        double time=scan.nextDouble();  
  
        FinanceCalculator calculator=new FinanceCalculator();  
  
        System.out.println(calculator.calculateSimpleInterest(principal, rate,  
time));  
  
        scan.close();  
  
    }  
  
}
```

Output:

Enter the Principal as Real number

1000.0

Enter the Rate of Interest as Real number

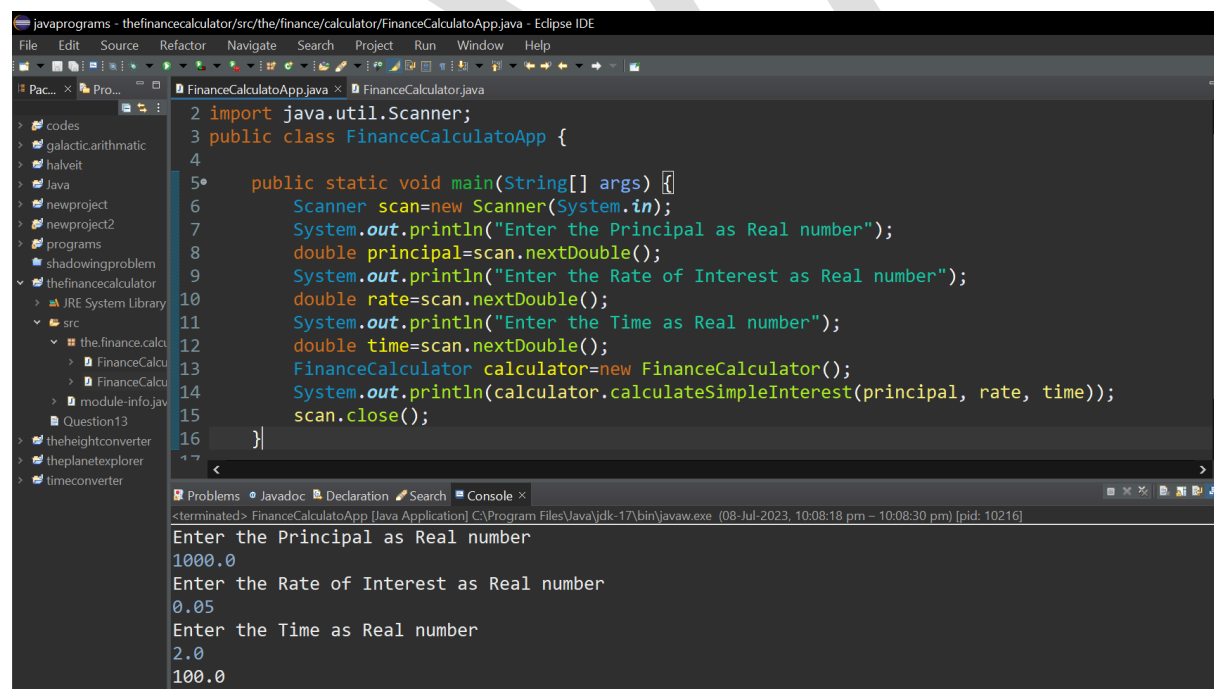
0.05

Enter the Time as Real number

2.0

100.0

Snapshot 13



The screenshot shows the Eclipse IDE with the file `FinanceCalcutoApp.java` open. The code is as follows:

```
1 import java.util.Scanner;
2 public class FinanceCalcutoApp {
3
4
5     public static void main(String[] args) {
6         Scanner scan=new Scanner(System.in);
7         System.out.println("Enter the Principal as Real number");
8         double principal=scan.nextDouble();
9         System.out.println("Enter the Rate of Interest as Real number");
10        double rate=scan.nextDouble();
11        System.out.println("Enter the Time as Real number");
12        double time=scan.nextDouble();
13        FinanceCalculator calculator=new FinanceCalculator();
14        System.out.println(calculator.calculateSimpleInterest(principal, rate, time));
15        scan.close();
16    }
17 }
```

The console output at the bottom shows the program's execution:

```
<terminated> FinanceCalcutoApp [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (08-Jul-2023, 10:08:18 pm - 10:08:30 pm) [pid: 10216]
Enter the Principal as Real number
1000.0
Enter the Rate of Interest as Real number
0.05
Enter the Time as Real number
2.0
100.0
```

Question: 14 **Problem: Time Converter**

You are developing a time tracking application for a company. The employees log their time in minutes. The management wants to see these durations in hours for better understanding. Your task is to write a function that can convert minutes into hours.

****Function:****

```
public static double convertToHours(int minutes);
```

****Inputs:****

The function convertToHours(int minutes) will receive one parameter:

- minutes ($1 \leq \text{minutes} \leq 10^6$): an integer which represents the number of minutes to be converted.

****Outputs:****

The function will return a double - the equivalent number of hours.

****Example:****

****Sample Input:****

```
convertToHours(90);
```

****Sample Output:****

1.5

****Note:****

In the sample input, the number of minutes given to the function is 90. The equivalent in hours is $90/60=1.5$, so the function returns 1.5 as the output. Your function should work accurately to provide correct information to the management.

Solution 14:

```
public class TimeConverter {  
    public static double convertToHours(int minutes) {  
        return minutes/60.0;  
    }  
}  
  
import java.util.Scanner;  
  
public class TimeConverterApp {  
  
    public static void main(String[] args) {
```

```

Scanner scan=new Scanner(System.in);

System.out.println("Enter the minutes: ");

int minutes=scan.nextInt();

System.out.println(TimeConverter.convertToHours(minutes));

scan.close();

}

}

```

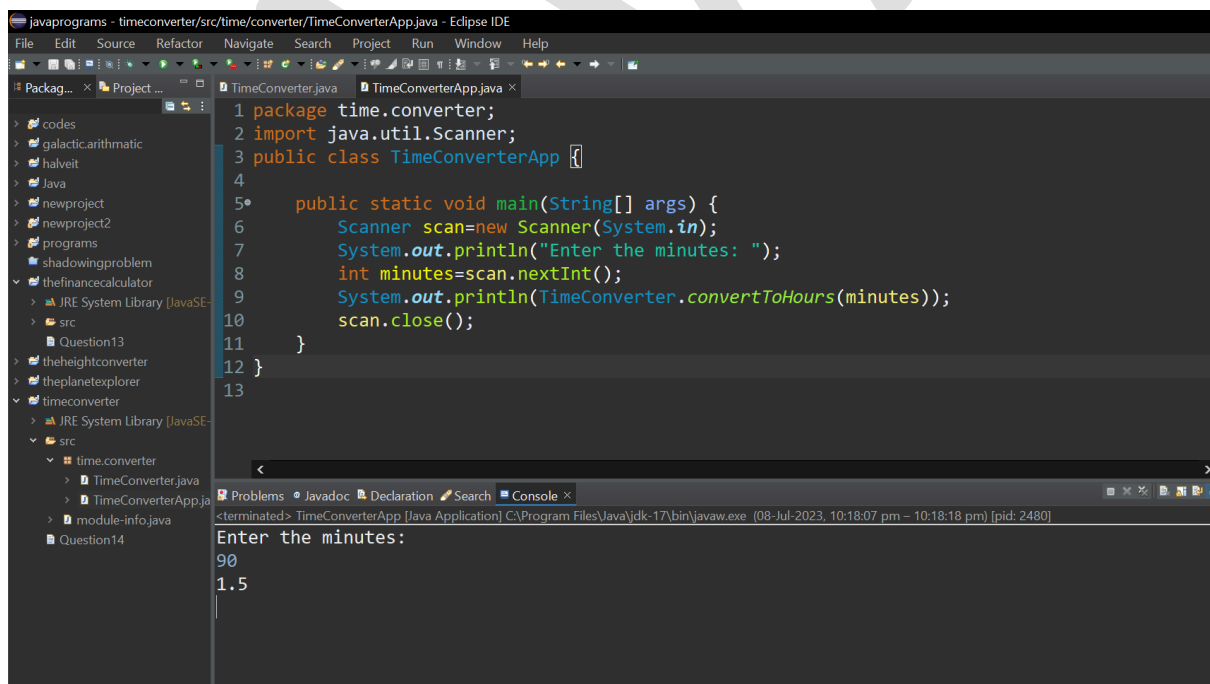
Output:

Enter the minutes:

90

1.5

Snapshot 14



Question: 15 **Problem: Halve It**

You are helping a friend in developing a financial app. The app has a feature where it calculates half of the entered amount for splitting bills. Your task is to write a function that takes a number and returns its half.

****Function:****

```
public static double halveTheNumber(double num);
```

****Inputs:****

The function `halveTheNumber(double num)` will receive one parameter:

- num ($0 \leq \text{num} \leq 10^9$): a double which represents the amount entered by the user to be halved.

****Outputs:****

The function will return a double - the result of halving the num.

****Example:****

****Sample Input:**

```
halveTheNumber(150.00);
```

Sample Output:

75.00

Note:

In the sample input, the number given to the function is 150.00. The half of this number is $150.00/2=75.00$, so the function returns 75.00 as the output. Make sure your function works correctly to split the bills accurately.

Solution 15:

```
public class HalfOfTheNumber {  
    public static double halveTheNumber(double num) {  
        return num/2;  
    }  
}
```

```
import java.util.Scanner;
```

```
public class HalfOfTheNumberApp {
```

```
public static void main(String[] args) {  
  
    Scanner scan=new Scanner(System.in);  
  
    System.out.println("Enter the number to be Halved as Real Num:");  
  
    double num=scan.nextDouble();  
  
    System.out.println(HalfOfTheNumber.halveTheNumber(num));  
  
    scan.close();  
  
}  
  
}
```

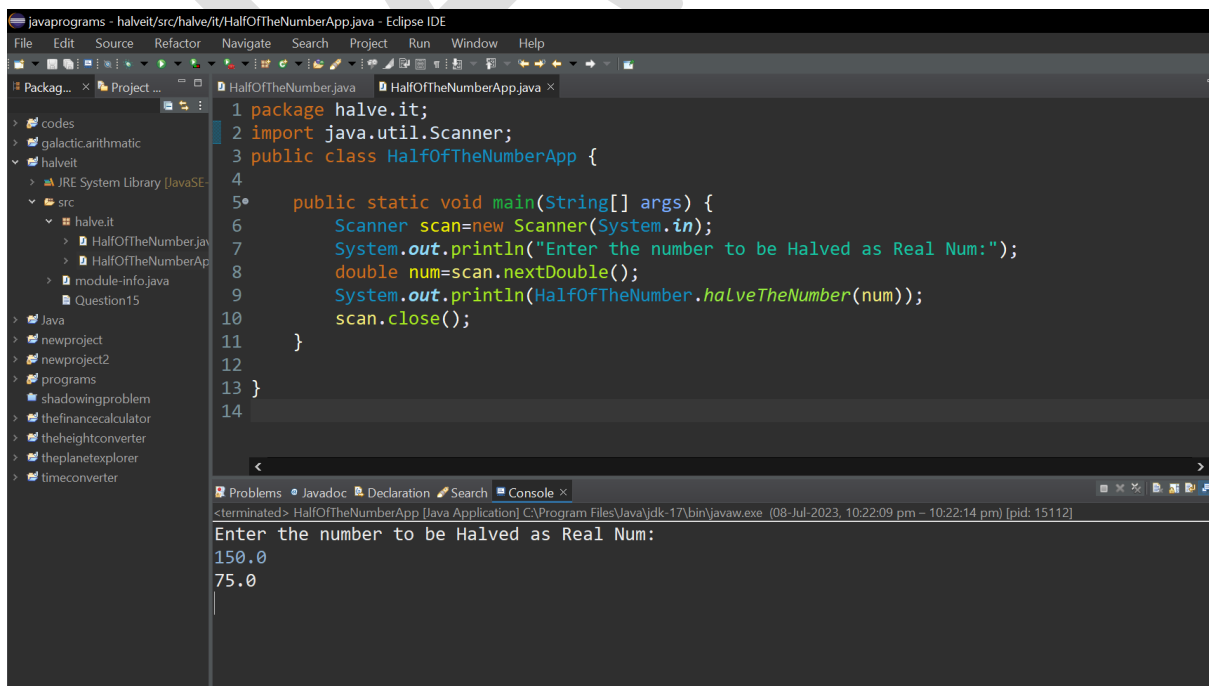
Output:

Enter the number to be Halved as Real Num:

150.0

75.0

Snapshot 15:



The screenshot shows the Eclipse IDE interface. The main editor displays the Java code for `HalfOfTheNumberApp.java`. The code is as follows:

```
1 package halve.it;  
2 import java.util.Scanner;  
3 public class HalfOfTheNumberApp {  
4  
5     public static void main(String[] args) {  
6         Scanner scan=new Scanner(System.in);  
7         System.out.println("Enter the number to be Halved as Real Num:");  
8         double num=scan.nextDouble();  
9         System.out.println(HalfOfTheNumber.halveTheNumber(num));  
10        scan.close();  
11    }  
12  
13 }  
14
```

The console output at the bottom shows the program execution:

```
<terminated> HalfOfTheNumberApp [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (08-Jul-2023, 10:22:09 pm - 10:22:14 pm) [pid: 15112]  
Enter the number to be Halved as Real Num:  
150.0  
75.0
```

JAVA