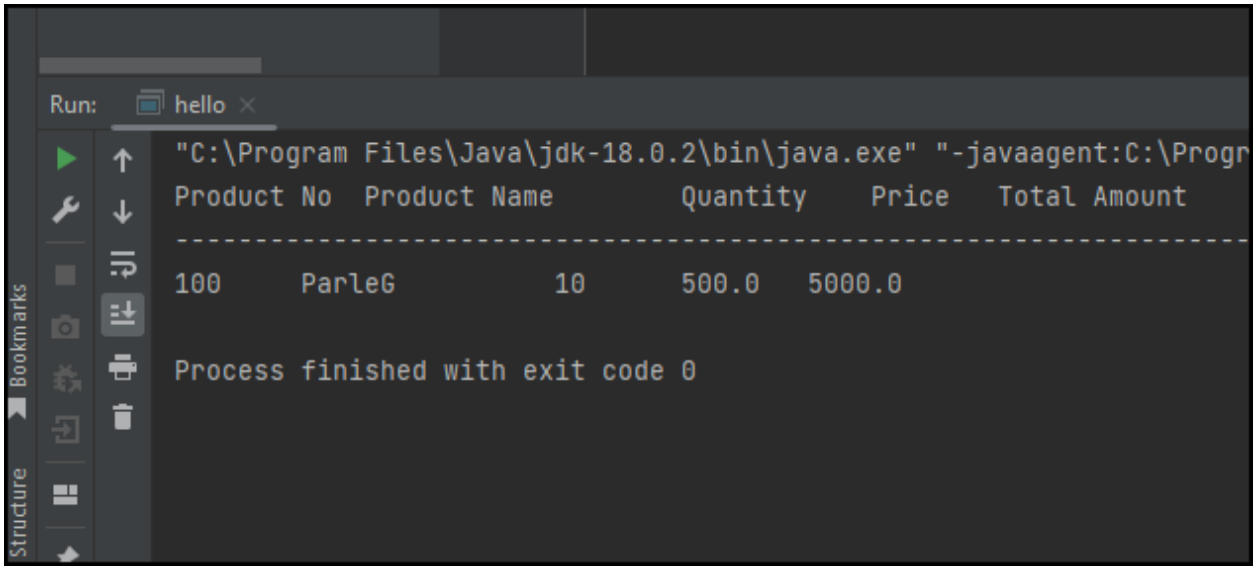
1. Create a class for a Product with product no(int), productname(String), price(float) and quantity(int) as attributes. Create a method to initialize variables (attributes) and another method to invoke values of attributes. Finally print the output in the console in the following format.

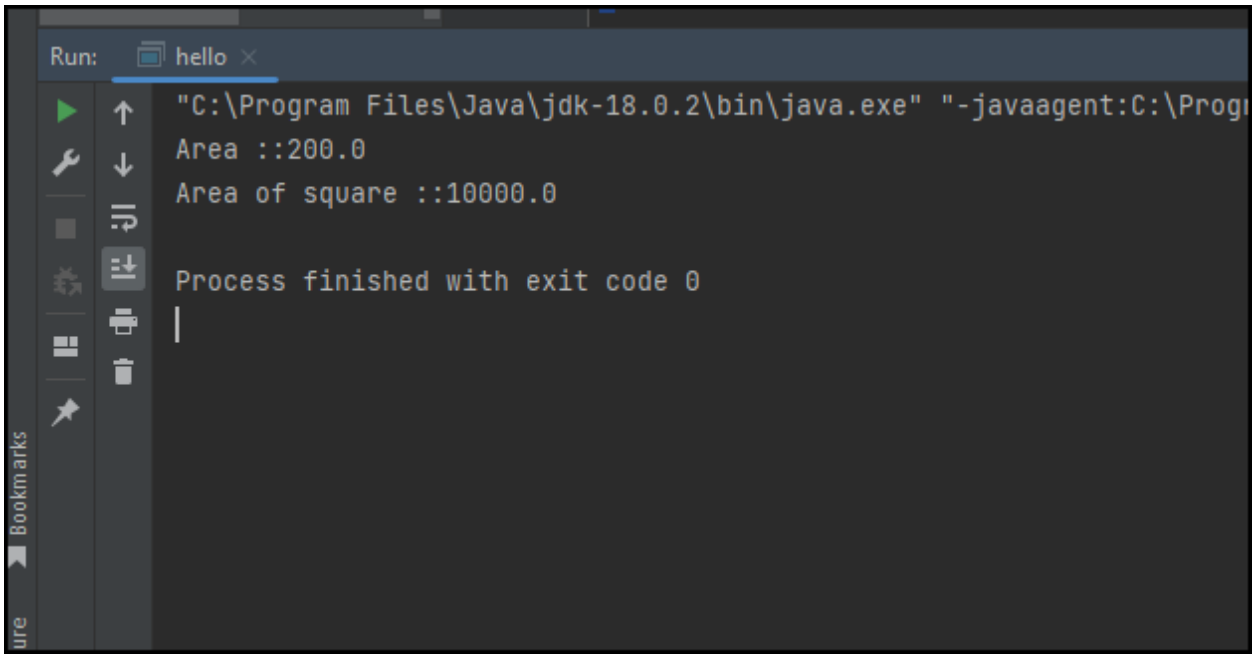
Product No	Product Name	Quantity	Price	Total Amount
Value	value	value	value	value

```
package com.example.hello;
import java.util.*;
import java.lang.*  ;
import java.util.Scanner;
class Product{
   int ProductNo , Quantity ;
   String ProductName ;
   float Price , TotalAmount ;
   void getProduct(int ProductNo , String ProductName , int Quantity
, float Price)
   {
       this.ProductNo = ProductNo ;
       this.ProductName = ProductName ;
       this.Quantity = Quantity ;
       this.Price  = Price ;
       TotalAmount = Quantity*Price ;
       this.TotalAmount = TotalAmount ;
   }
   void setProduct()
   {
       System.out.println(ProductNo + "\t\t" + ProductName +
"\t\t\t" + Quantity + "\t\t" + Price + "\t"  + TotalAmount);
   }
}
public class hello {
   public static void main(String[] args) {
       System.out.println("Product No\t" + "Product Name\t\t" +
"Quantity\t" + "Price\t" + "Total Amount");
System.out.println("-------------------------------------------------
---------------------------------");
       Product p = new Product() ;
       p.getProduct(100 , "ParleG" , 10 , 500);
       p.setProduct();
   }
}
```



2.Write a program to calculate the area of a rectangle and the area of a square by using method overloading. Take the input for dimension of rectangle and square from console. Handle appropriate exceptions in order to stop abnormal termination of execution.

```
package com.example.hello;
import java.util.*;
import java.lang.*  ;

// sout-- printing single lines
// psvm --> main file

import java.util.Scanner;

class Overloading{
   void Area(double  l , double b ){
       System.out.println("Area ::"  + l*b);
   }

   void Area(double side)
   {
       System.out.println("Area of square ::" +  side*side);
   }
}
public class hello {

   public static void main(String[] args) {

       Overloading o =  new Overloading() ;
       o.Area(10  , 20);
       o.Area(100);    }}
```



3. Create an abstract class called Employee with following attributes and methods.

Empid(int), empname(String) -> Attributes

input(int,String) -> Concrete Method

output() -> Concrete Method

roles_Responsibilities() -> abstract method

Create a class called Manager which is a subclass of Employee and override the abstract method of Employee. Invoke all the methods.

(Note: Implementation of methods should be appropriate)

```
package com.example.hello;
import java.util.*;
import java.lang.*  ;
//vineet codes java
// sout-- printing single lines
// psvm --> main file
```

```java
import java.util.Scanner;
abstract class Employee {
   String empName  ;
   private int empId;
   abstract public  void role_Responsibilities() ;
   public void setInput(int id , String name )
   {
       empId = id ;
       empName = name ;
   }
   public void getInput(){
       System.out.println("emp id ::" +  empId);
       System.out.println("emp name ::" + empName);
   }
}
class Manager extends  Employee{
   @Override
   public void role_Responsibilities() {
       System.out.println("The role of manager and responsibility wil be
xyz...");
   }
}
public class hello {
   public static void main(String[] args) {
Manager m = new Manager()  ;
m.setInput(100 , "vineet");
m.getInput();
m.role_Responsibilities();
   }
}
```