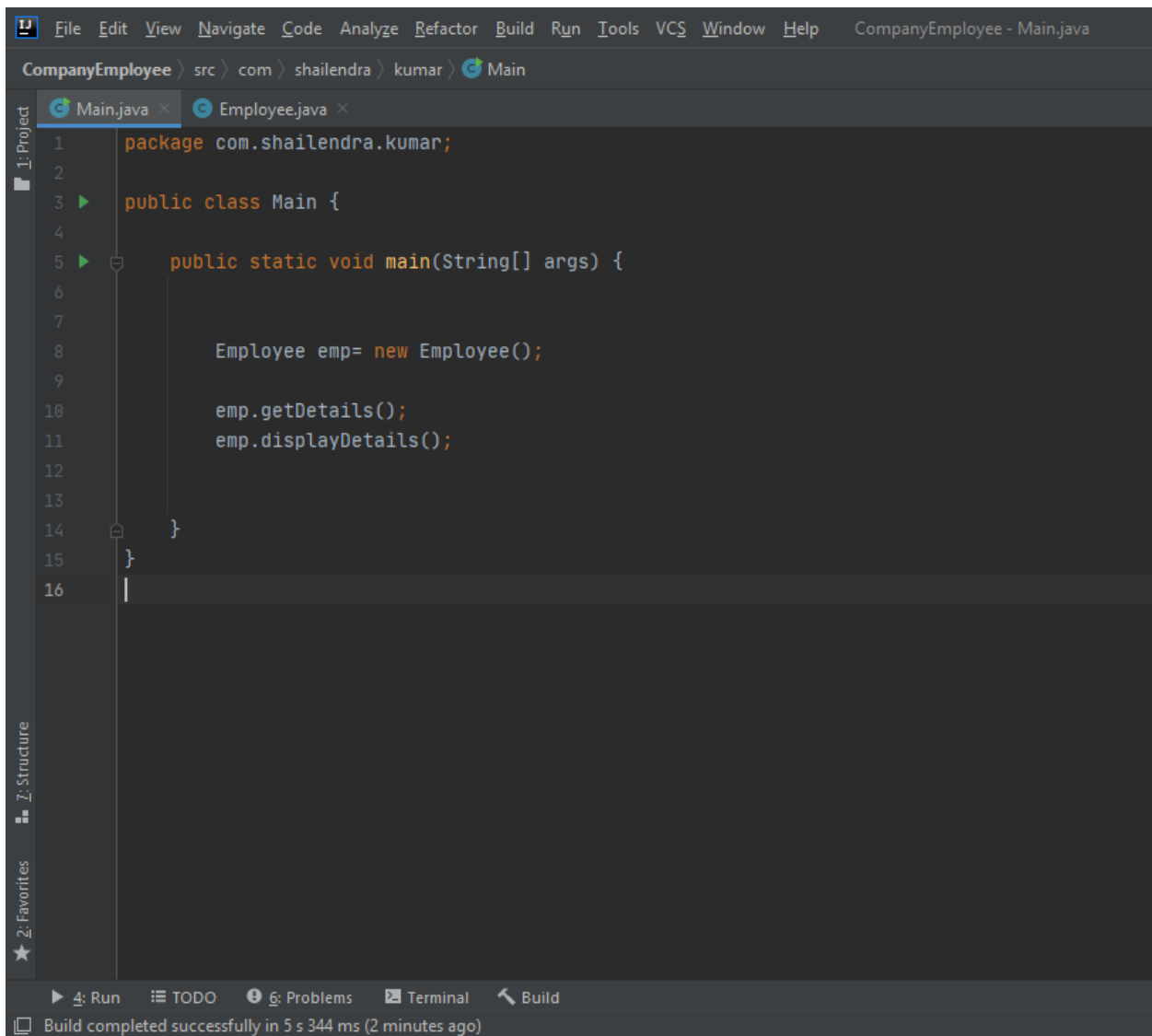


LetsUpgrade Java Essentials Assignment - 5

Questions 1:

Create a parent class for different employees with common details name, age, salary, designation, and methods to get details and display details.

Answer:

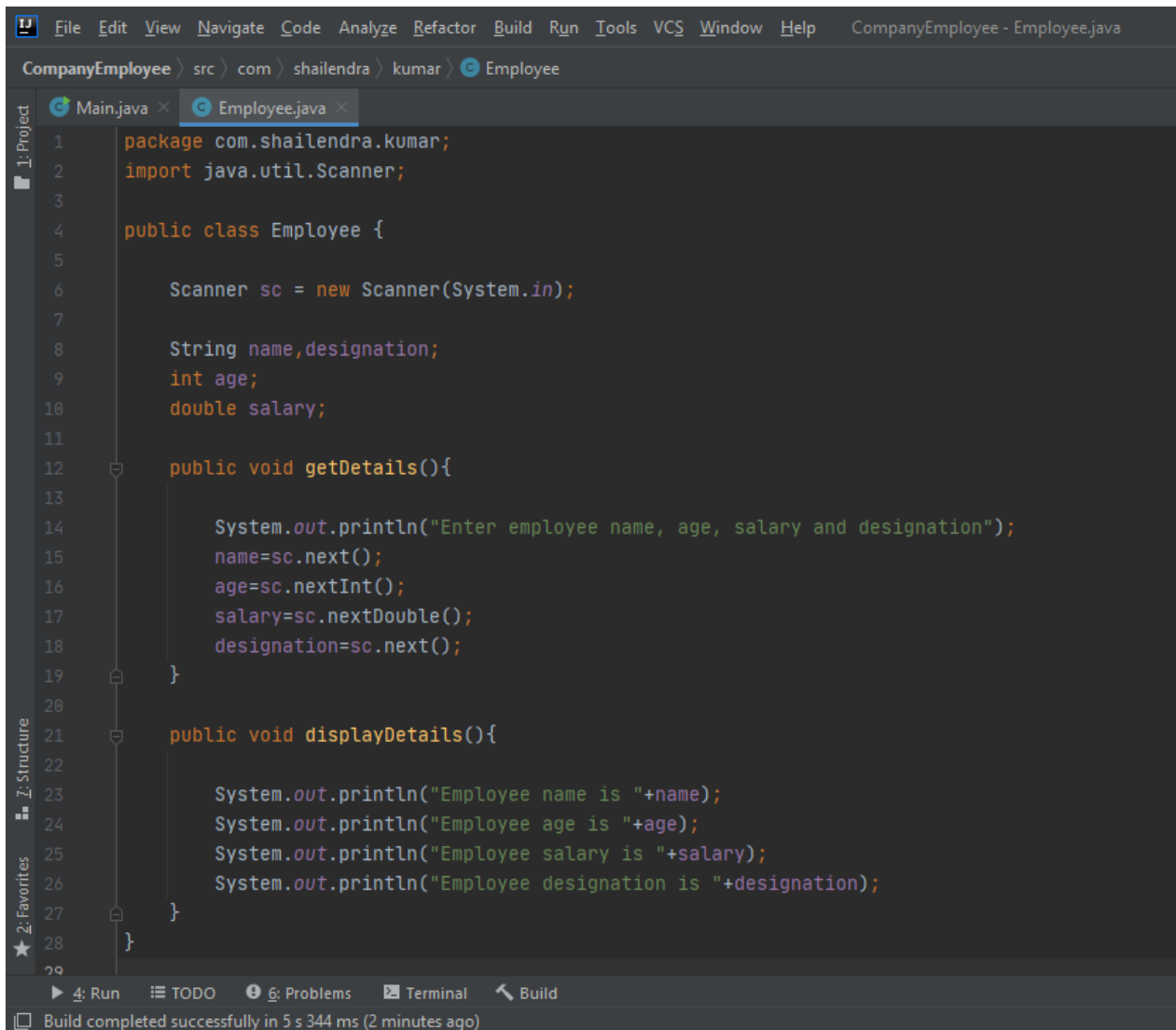


```
1 package com.shailendra.kumar;
2
3 public class Main {
4
5     public static void main(String[] args) {
6
7         Employee emp= new Employee();
8
9         emp.getDetails();
10        emp.displayDetails();
11
12    }
13
14 }
15
16 |
```

The screenshot shows an IDE window titled 'CompanyEmployee - Main.java'. The breadcrumb navigation shows the path 'CompanyEmployee > src > com > shailendra > kumar > Main'. The editor displays the 'Main.java' file with the following code:

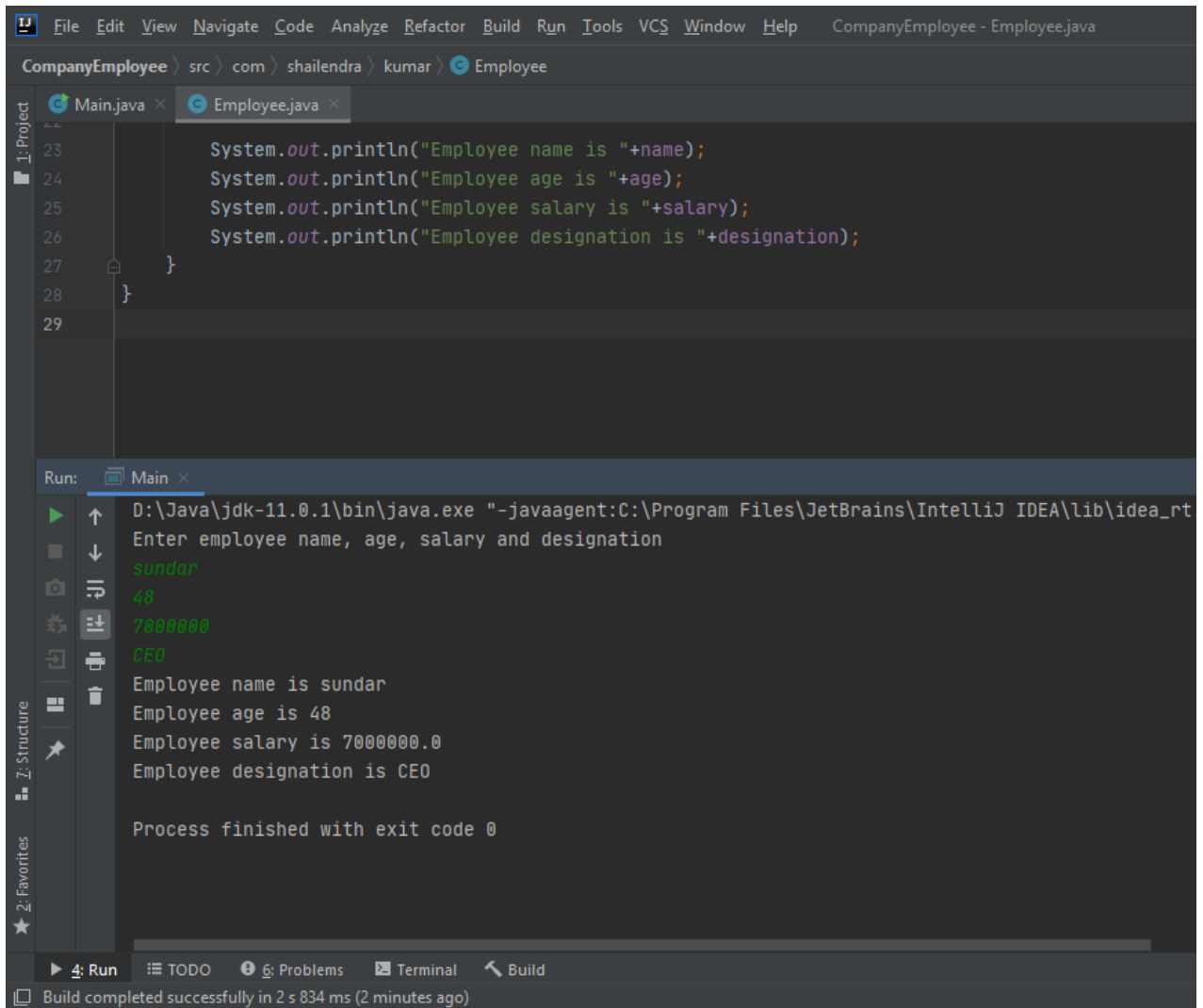
```
1 package com.shailendra.kumar;
2
3 public class Main {
4
5     public static void main(String[] args) {
6
7         Employee emp= new Employee();
8
9         emp.getDetails();
10        emp.displayDetails();
11
12    }
13
14 }
15
16 |
```

The IDE interface includes a menu bar (File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, Help), a sidebar with 'Project', 'Structure', and 'Favorites' views, and a bottom status bar showing 'Build completed successfully in 5 s 344 ms (2 minutes ago)'.



```
1 package com.shailendra.kumar;
2 import java.util.Scanner;
3
4 public class Employee {
5
6     Scanner sc = new Scanner(System.in);
7
8     String name, designation;
9     int age;
10    double salary;
11
12    public void getDetails(){
13
14        System.out.println("Enter employee name, age, salary and designation");
15        name=sc.next();
16        age=sc.nextInt();
17        salary=sc.nextDouble();
18        designation=sc.next();
19    }
20
21    public void displayDetails(){
22
23        System.out.println("Employee name is "+name);
24        System.out.println("Employee age is "+age);
25        System.out.println("Employee salary is "+salary);
26        System.out.println("Employee designation is "+designation);
27    }
28 }
29
```

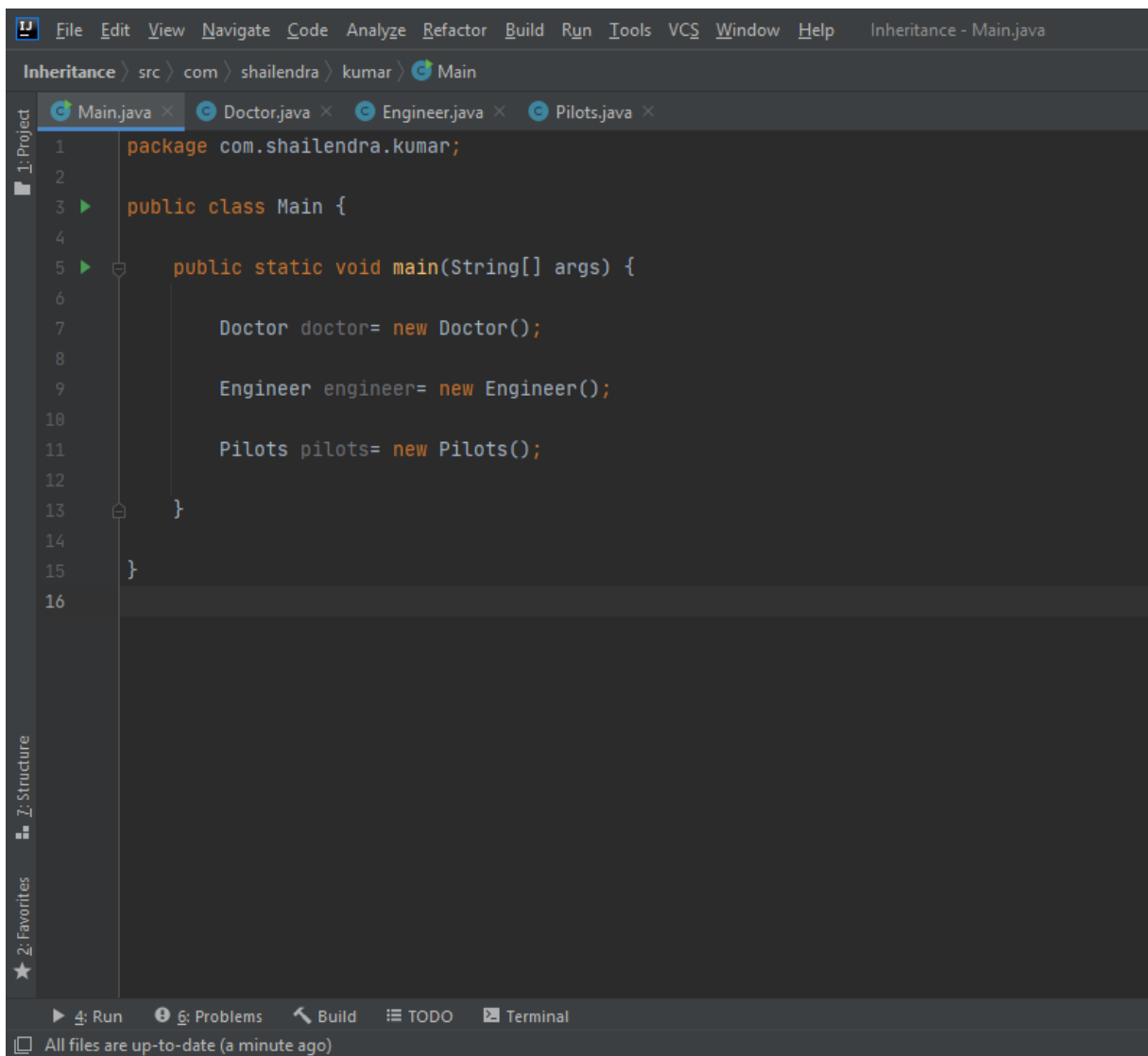
Build completed successfully in 5 s 344 ms (2 minutes ago)



Questions 2:

Create three child classes one for the doctor, one for an engineer, and one for pilots and include their different operations in respective classes.

Answer:



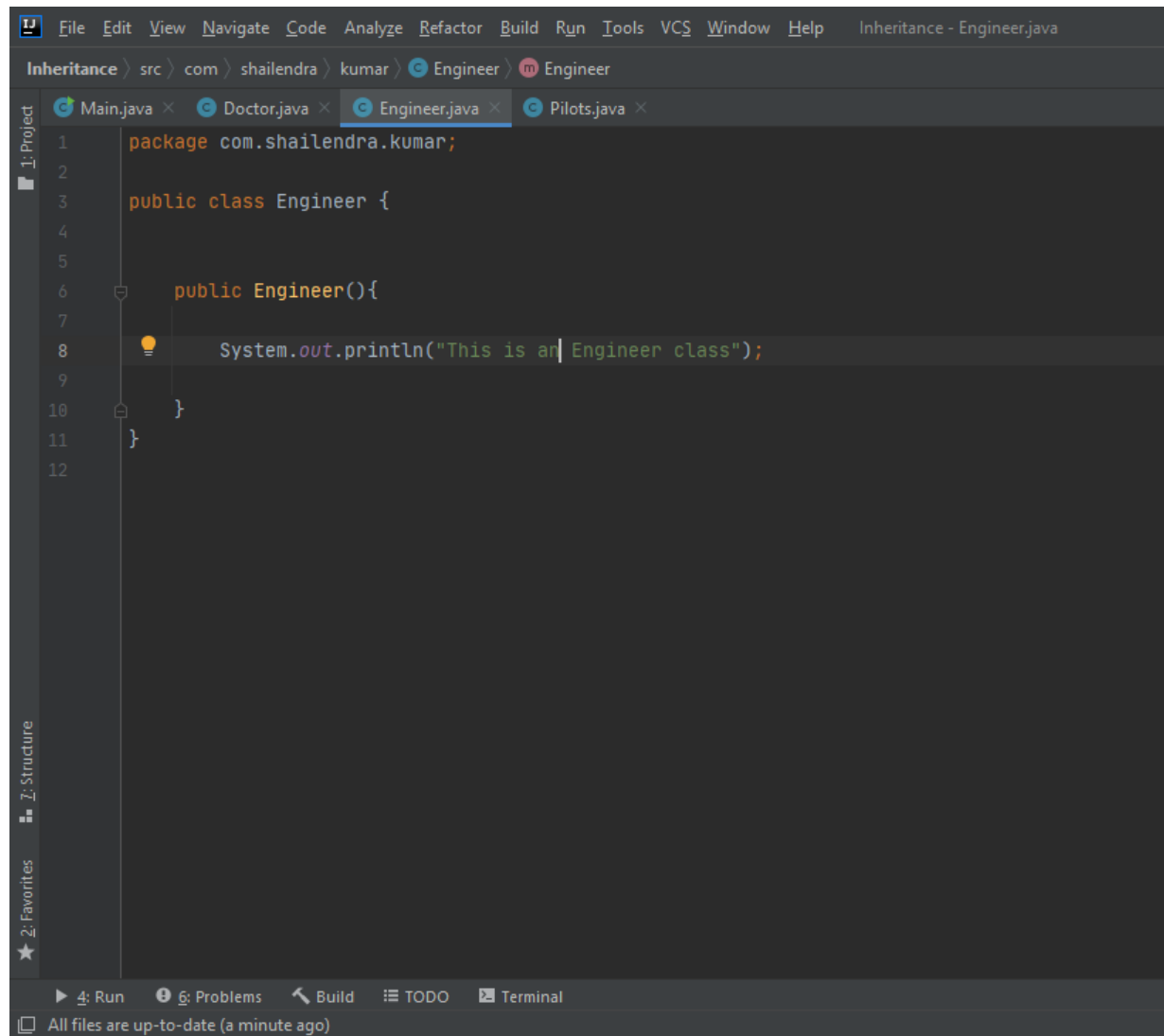
The screenshot shows an IDE window titled "Inheritance - Main.java". The package path is "src > com > shailendra > kumar". The code defines a "Main" class with a "main" method that instantiates "Doctor", "Engineer", and "Pilots" objects. The IDE interface includes a menu bar, a toolbar, a project explorer on the left, and a status bar at the bottom.

```
1 package com.shailendra.kumar;
2
3 public class Main {
4
5     public static void main(String[] args) {
6
7         Doctor doctor= new Doctor();
8
9         Engineer engineer= new Engineer();
10
11         Pilots pilots= new Pilots();
12
13     }
14
15 }
16
```

The screenshot shows an IDE window titled "Inheritance - Doctor.java". The breadcrumb navigation shows the path: `Inheritance > src > com > shailendra > kumar > Doctor`. The editor has four tabs: `Main.java`, `Doctor.java` (active), `Engineer.java`, and `Pilots.java`. The code in `Doctor.java` is as follows:

```
1 package com.shailendra.kumar;
2
3 public class Doctor {
4
5
6     public Doctor(){
7
8         System.out.println("This is a Doctor class");
9
10    }
11
12 }
13
```

The left sidebar contains three views: "1: Project", "7: Structure", and "2: Favorites". The bottom status bar shows icons for "4: Run", "6: Problems", "Build", "TODO", and "Terminal", along with the text "All files are up-to-date (a minute ago)".



The screenshot shows an IDE window titled "Inheritance - Engineer.java". The breadcrumb navigation shows the path: `src > com > shailendra > kumar > Engineer > Engineer`. The editor displays the following Java code:

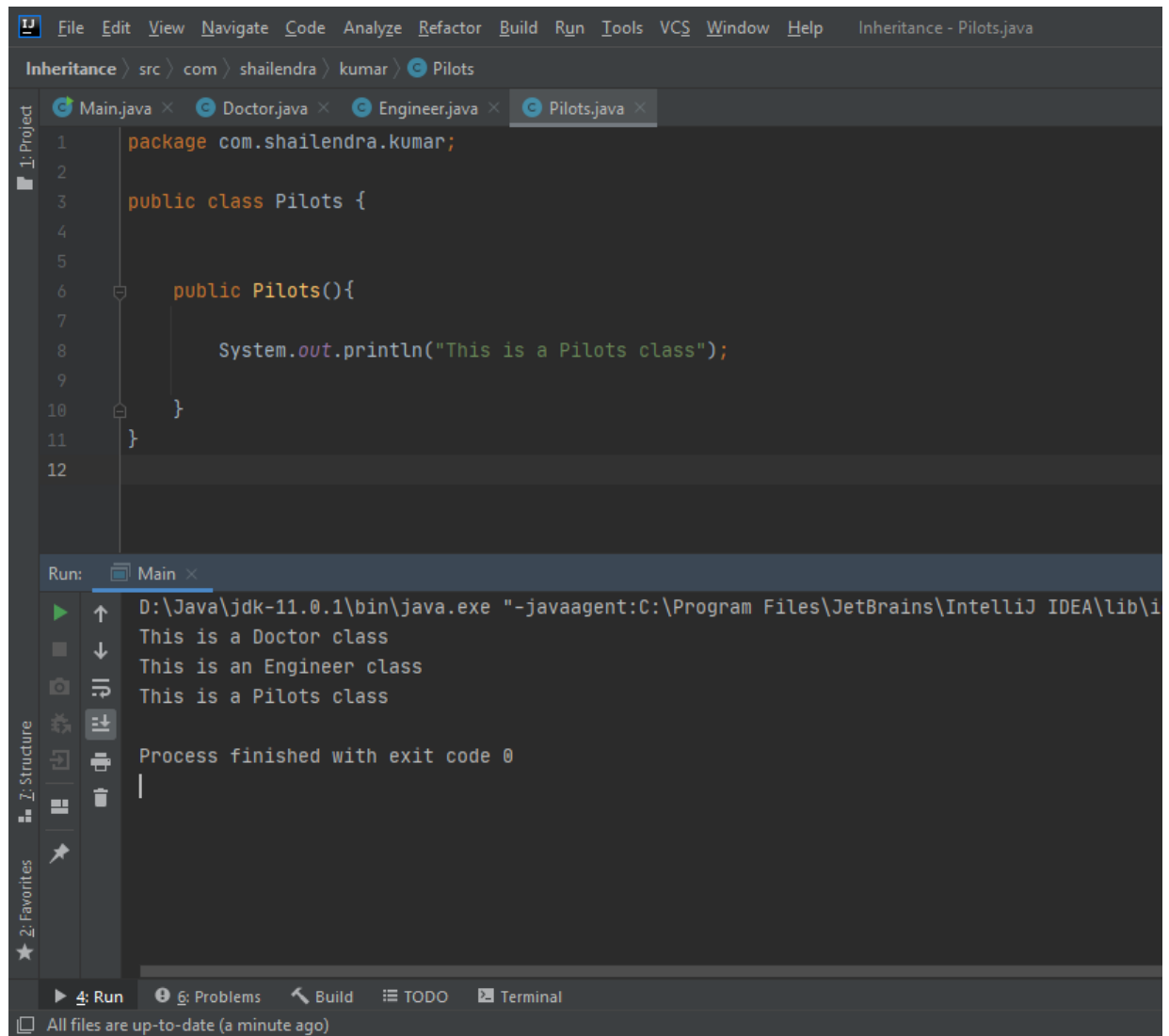
```
1 package com.shailendra.kumar;
2
3 public class Engineer {
4
5     public Engineer(){
6
7         System.out.println("This is an Engineer class");
8     }
9 }
10
11
12
```

The left sidebar contains three tabs: "1: Project", "2: Structure", and "3: Favorites". The bottom status bar shows icons for "Run", "Problems", "Build", "TODO", and "Terminal", along with the text "All files are up-to-date (a minute ago)".

The screenshot shows an IDE window titled "Inheritance - Pilots.java". The menu bar includes File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, and Help. The breadcrumb navigation shows the path: Inheritance > src > com > shailendra > kumar > Pilots. The tab bar shows four open files: Main.java, Doctor.java, Engineer.java, and Pilots.java (which is the active file). The left sidebar has three sections: "1: Project" (showing a file tree), "2: Structure" (showing a class hierarchy), and "3: Favorites" (showing a star icon). The main editor area displays the following Java code:

```
1 package com.shailendra.kumar;
2
3 public class Pilots {
4
5
6     public Pilots(){
7
8         System.out.println("This is a Pilots class");
9
10    }
11 }
12
```

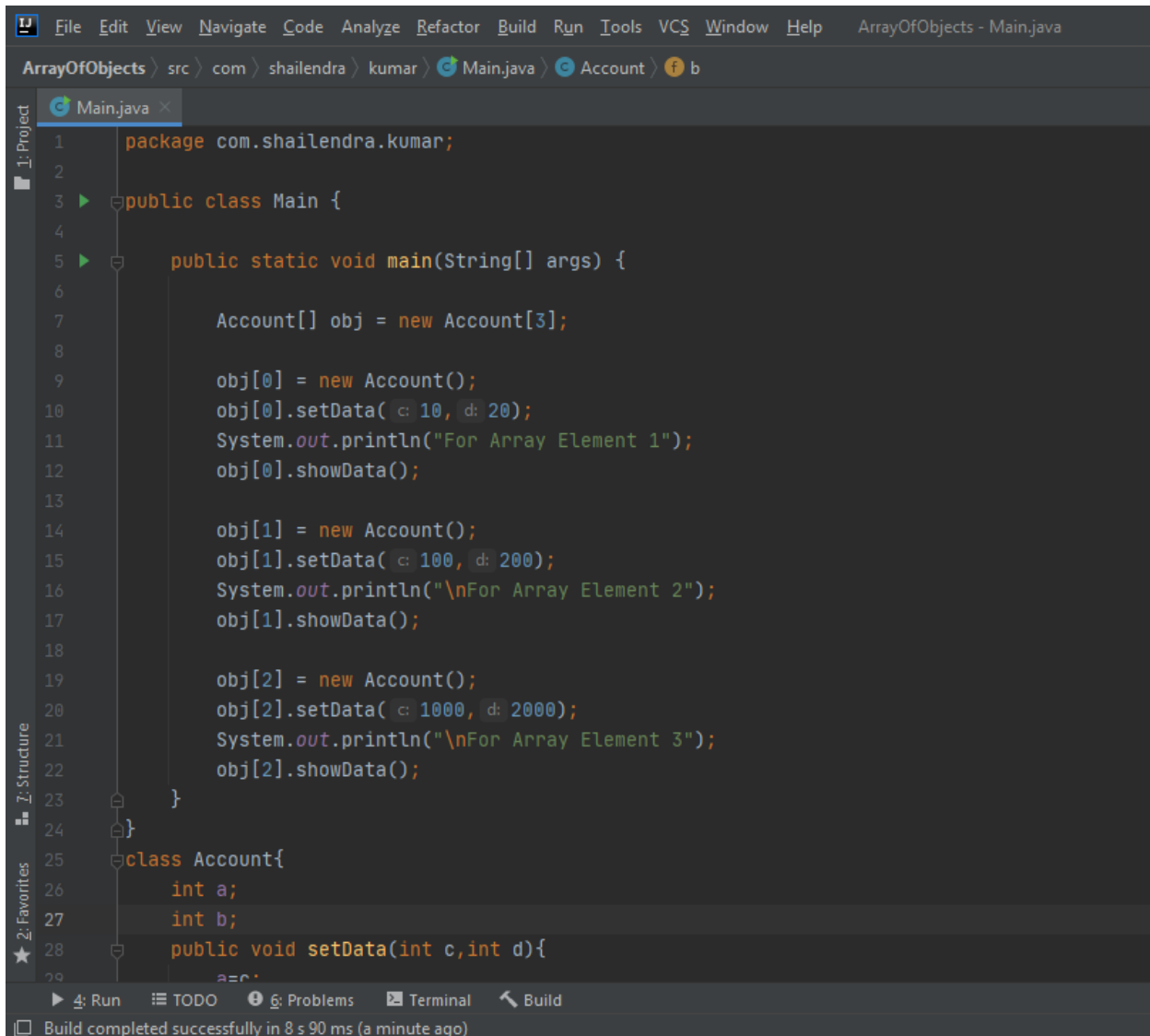
The bottom status bar shows icons for Run, Problems, Build, TODO, and Terminal, along with the text "All files are up-to-date (a minute ago)".



Questions 3:

In the main method create an array of objects for three of them get the details and print for all the objects.

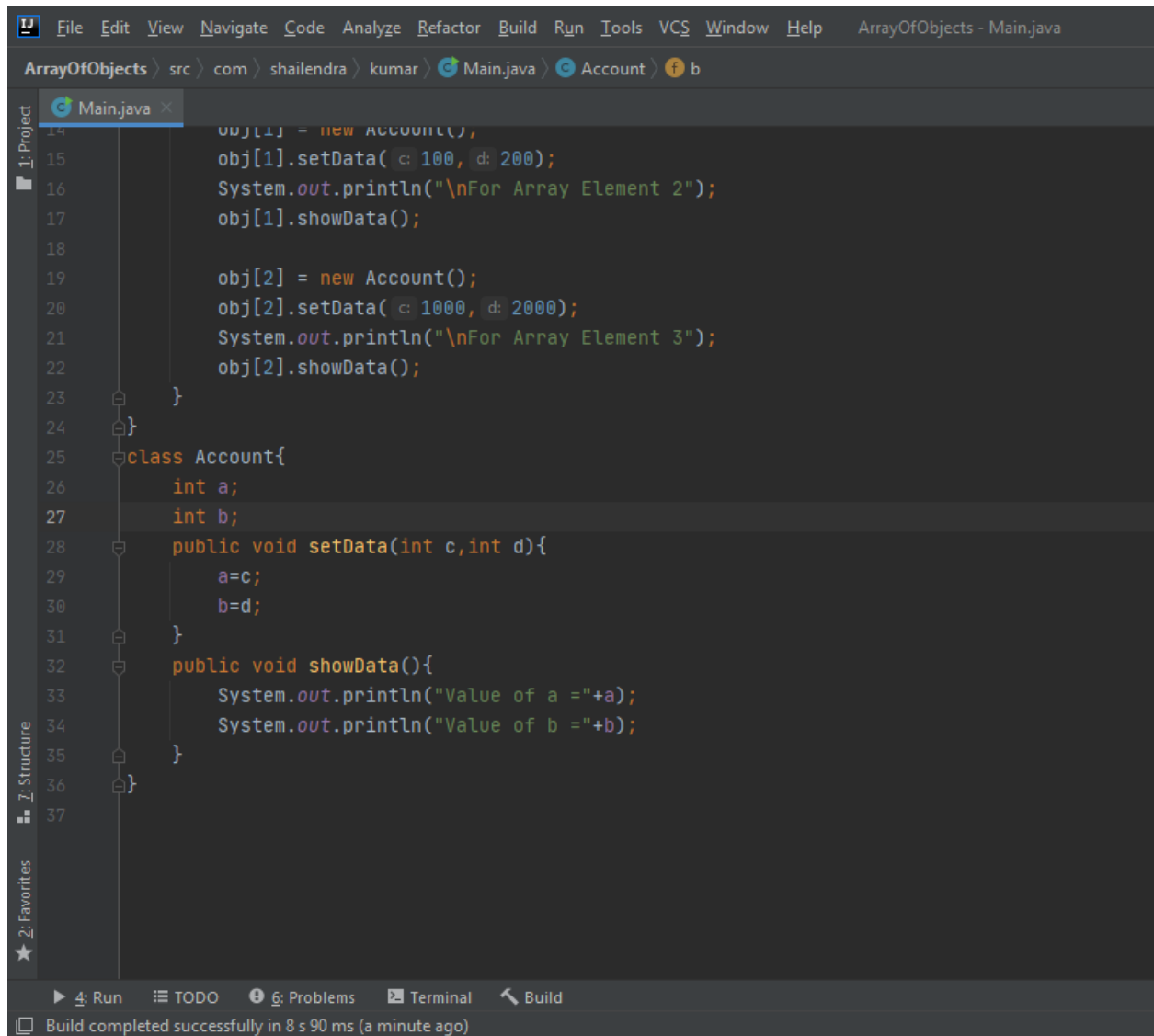
Answer:



```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help ArrayOfObjects - Main.java
ArrayOfObjects > src > com > shailendra > kumar > Main.java > Account > b
Main.java X
1 package com.shailendra.kumar;
2
3 public class Main {
4
5     public static void main(String[] args) {
6
7         Account[] obj = new Account[3];
8
9         obj[0] = new Account();
10        obj[0].setData( c: 10, d: 20);
11        System.out.println("For Array Element 1");
12        obj[0].showData();
13
14        obj[1] = new Account();
15        obj[1].setData( c: 100, d: 200);
16        System.out.println("\nFor Array Element 2");
17        obj[1].showData();
18
19        obj[2] = new Account();
20        obj[2].setData( c: 1000, d: 2000);
21        System.out.println("\nFor Array Element 3");
22        obj[2].showData();
23    }
24 }
25 class Account{
26     int a;
27     int b;
28     public void setData(int c,int d){
29         a=c;
30     }
31 }
```

Run TODO Problems Terminal Build

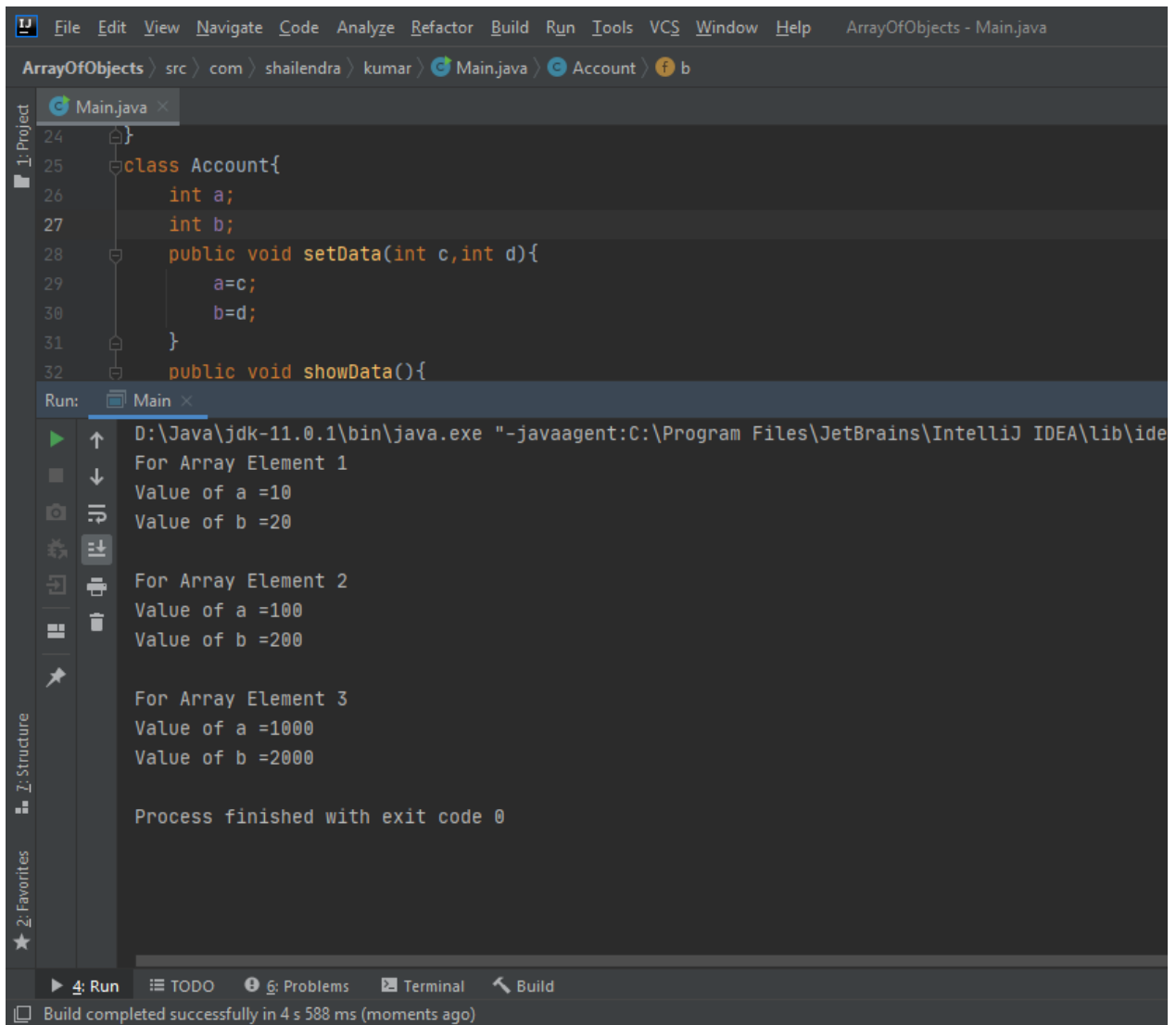
Build completed successfully in 8 s 90 ms (a minute ago)



The screenshot shows an IDE window titled "ArrayOfObjects - Main.java". The breadcrumb navigation shows the path: "ArrayOfObjects > src > com > shailendra > kumar > Main.java > Account > b". The editor displays the following Java code:

```
14 obj[1] = new Account();
15 obj[1].setData(c: 100, d: 200);
16 System.out.println("\nFor Array Element 2");
17 obj[1].showData();
18
19 obj[2] = new Account();
20 obj[2].setData(c: 1000, d: 2000);
21 System.out.println("\nFor Array Element 3");
22 obj[2].showData();
23 }
24 }
25 class Account{
26     int a;
27     int b;
28     public void setData(int c,int d){
29         a=c;
30         b=d;
31     }
32     public void showData(){
33         System.out.println("Value of a =" +a);
34         System.out.println("Value of b =" +b);
35     }
36 }
37
```

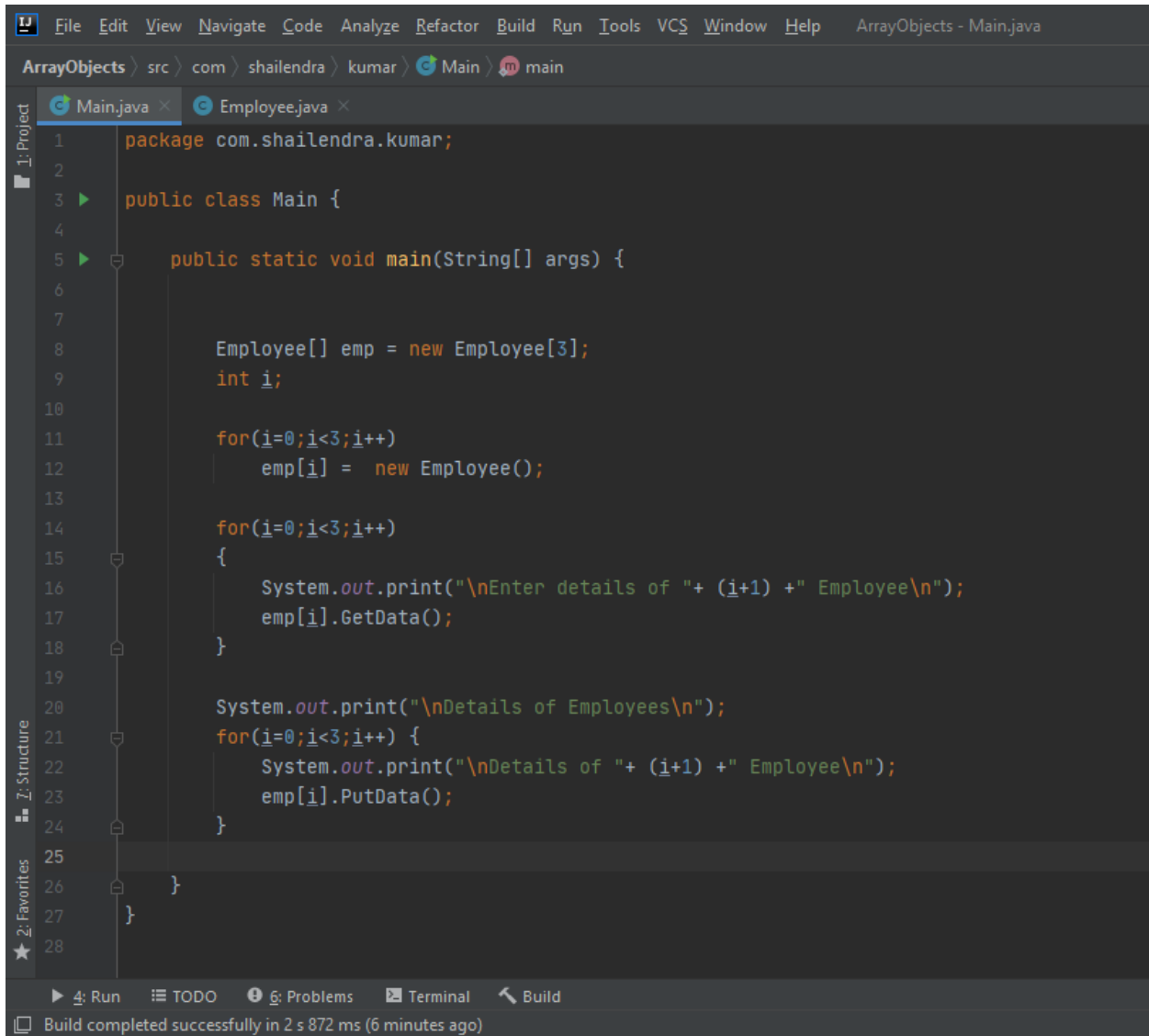
The left sidebar shows the "Project" view with "Main.java" selected, and the "Structure" view showing the class hierarchy. The bottom status bar indicates "Build completed successfully in 8 s 90 ms (a minute ago)".



Questions 4:

Array of 3 objects for each type of employee.

Answer:

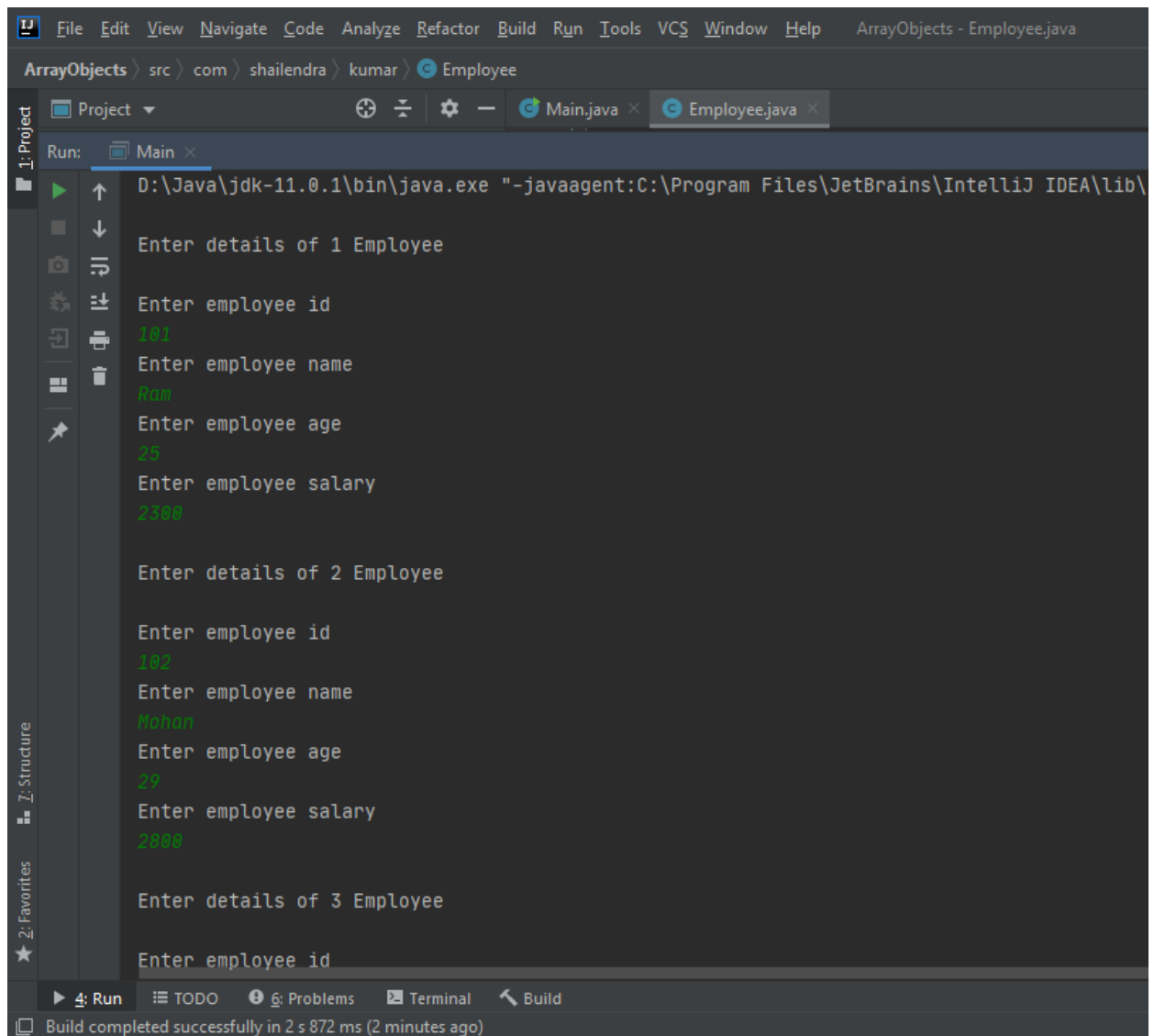


```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help ArrayObjects - Main.java
ArrayObjects > src > com > shailendra > kumar > Main > main
Main.java x Employee.java x
1 package com.shailendra.kumar;
2
3 public class Main {
4
5     public static void main(String[] args) {
6
7
8         Employee[] emp = new Employee[3];
9         int i;
10
11         for(i=0; i<3; i++)
12             emp[i] = new Employee();
13
14         for(i=0; i<3; i++)
15         {
16             System.out.print("\nEnter details of " + (i+1) + " Employee\n");
17             emp[i].GetData();
18         }
19
20         System.out.print("\nDetails of Employees\n");
21         for(i=0; i<3; i++) {
22             System.out.print("\nDetails of " + (i+1) + " Employee\n");
23             emp[i].PutData();
24         }
25
26     }
27 }
28
Run TODO Problems Terminal Build
Build completed successfully in 2 s 872 ms (6 minutes ago)
```

The screenshot shows an IDE window titled "ArrayObjects - Employee.java". The breadcrumb path is "ArrayObjects > src > com > shailendra > kumar > Employee". The editor displays the following Java code:

```
1 package com.shailendra.kumar;
2 import java.util.Scanner;
3
4 public class Employee {
5     int id;
6     String name;
7     int age;
8     double salary;
9     void GetData()
10    {
11        Scanner sc = new Scanner(System.in);
12        System.out.println("\nEnter employee id");
13        id=sc.nextInt();
14        System.out.println("Enter employee name");
15        name=sc.next();
16        System.out.println("Enter employee age");
17        age=sc.nextInt();
18        System.out.println("Enter employee salary");
19        salary=sc.nextDouble();
20    }
21    void PutData()
22    {
23        System.out.print("Employee id is "+id);
24        System.out.print("\nEmployee name is "+name);
25        System.out.print("\nEmployee age is "+age);
26        System.out.print("\nEmployee salary is "+salary+"\n");
27    }
28 }
29
```

The IDE interface includes a menu bar (File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, Help), a sidebar with "Project", "Structure", and "Favorites" views, and a bottom status bar showing "Build completed successfully in 2 s 872 ms (7 minutes ago)".



```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help ArrayObjects - Employee.java
ArrayObjects > src > com > shailendra > kumar > Employee
Project Main Employee.java
Run: Main
D:\Java\jdk-11.0.1\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA\lib\
Enter details of 1 Employee
Enter employee id
101
Enter employee name
Ram
Enter employee age
25
Enter employee salary
23000
Enter details of 2 Employee
Enter employee id
102
Enter employee name
Mohan
Enter employee age
29
Enter employee salary
28000
Enter details of 3 Employee
Enter employee id
103
Enter employee name
Anand
Enter employee age
35
Enter employee salary
35000
Build completed successfully in 2 s 872 ms (2 minutes ago)
```

```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help ArrayObjects - Employee.java
ArrayObjects > src > com > shailendra > kumar > Employee
Project
Run: Main
Enter details of 3 Employee
Enter employee id
103
Enter employee name
Syam
Enter employee age
26
Enter employee salary
3200
Details of Employees
Details of 1 Employee
Employee id is 101
Employee name is Ram
Employee age is 25
Employee salary is 2300.0
Details of 2 Employee
Employee id is 102
Employee name is Mohan
Employee age is 29
Employee salary is 2800.0
|
Details of 3 Employee
Employee id is 103
4: Run TODO 6: Problems Terminal Build
Build completed successfully in 2 s 872 ms (2 minutes ago)
```

```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help ArrayObjects - Employee.java
ArrayObjects > src > com > shailendra > kumar > Employee
Project Main.java Employee.java
Run: Main
Enter Employee age
25
Enter employee salary
3200
Details of Employees
Details of 1 Employee
Employee id is 101
Employee name is Ram
Employee age is 25
Employee salary is 2300.0
Details of 2 Employee
Employee id is 102
Employee name is Mohan
Employee age is 29
Employee salary is 2800.0
Details of 3 Employee
Employee id is 103
Employee name is Syam
Employee age is 26
Employee salary is 3200.0
Process finished with exit code 0
4: Run TODO 6: Problems Terminal Build
Build completed successfully in 2 s 872 ms (3 minutes ago)
```

By Shailendra Kumar

Part of LetsUpgrade Family