*LAB # 02*

arraylist and vector in java

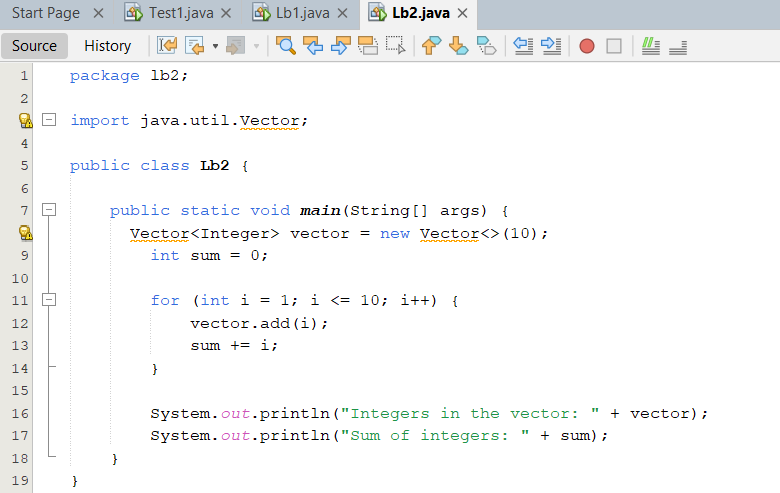
# *OBJECTIVE:*

*To implement ArayList and Vector.*

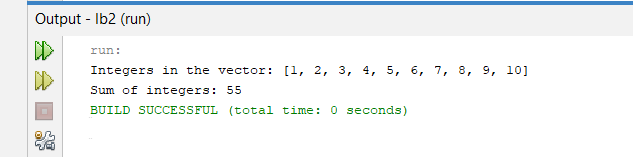
*LAB task*

1. *Write a program that initializes Vector with 10 integers in it. Display all the integers and sum of these integers.*

**Code:**

****

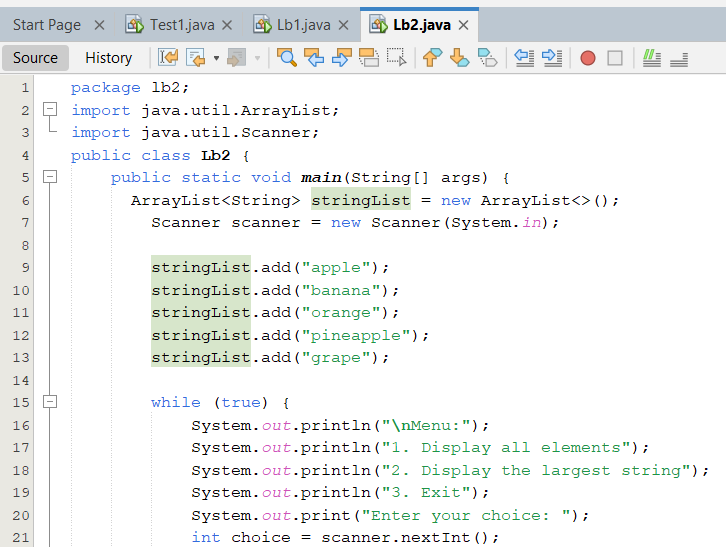
**Output:**

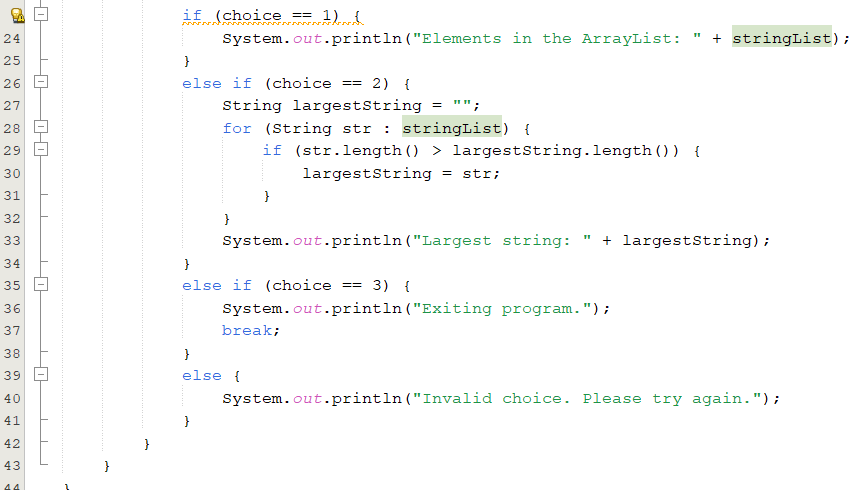
****

1. *Create a ArrayList of string. Write a menu driven program which:*

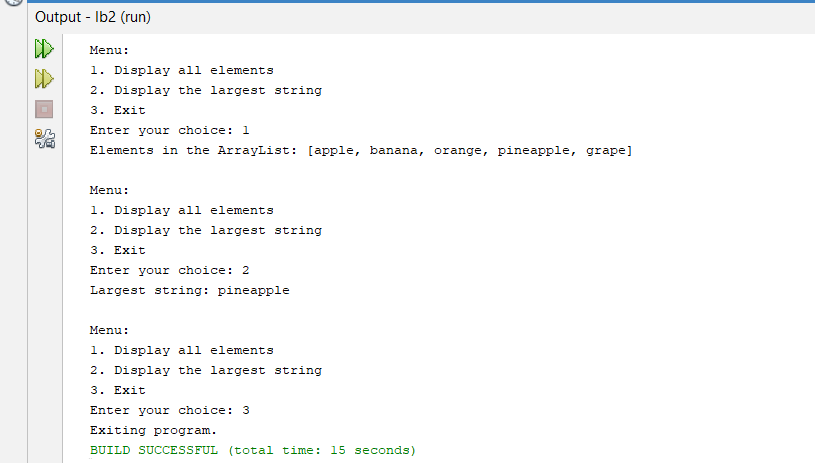
*a. Displays all the elements b. Displays the largest String*

**Code:**

****

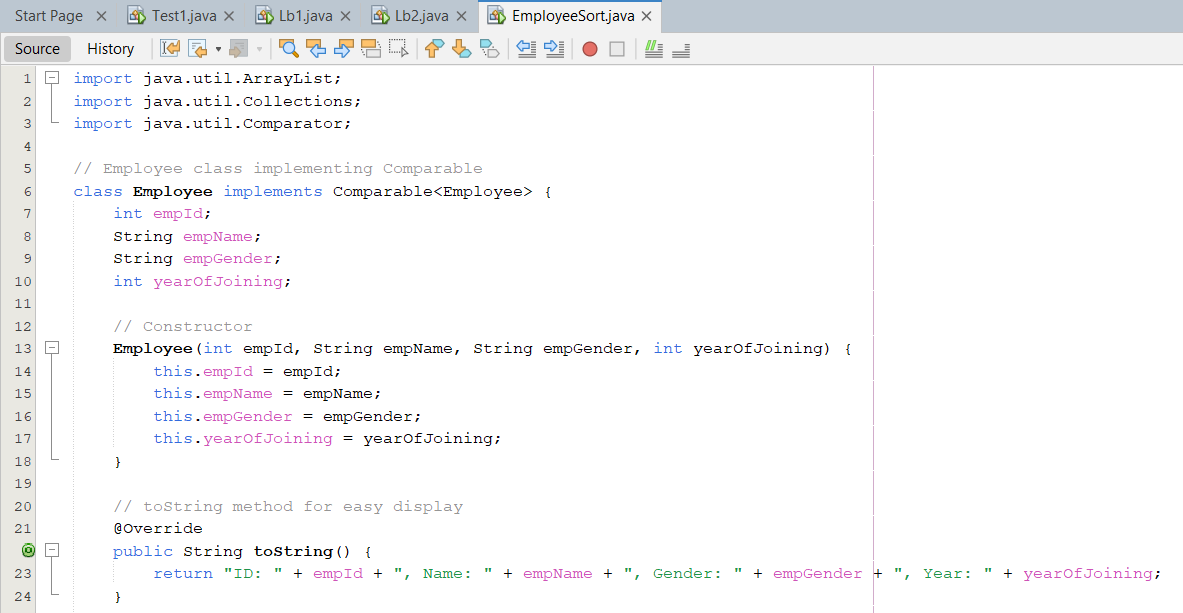
****

**Output:**

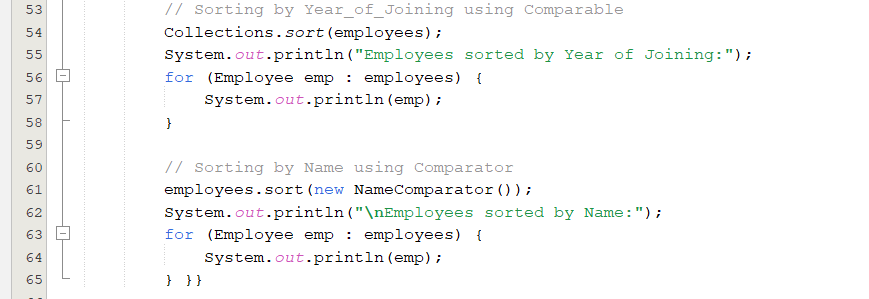
**

1. *Create a Arraylist storing Employee details including Emp\_id, Emp\_Name, Emp\_gender, Year\_of\_Joining (you can also add more attributes including these). Then sort the employees according to their joining year using Comparator and Comparable interfaces.*

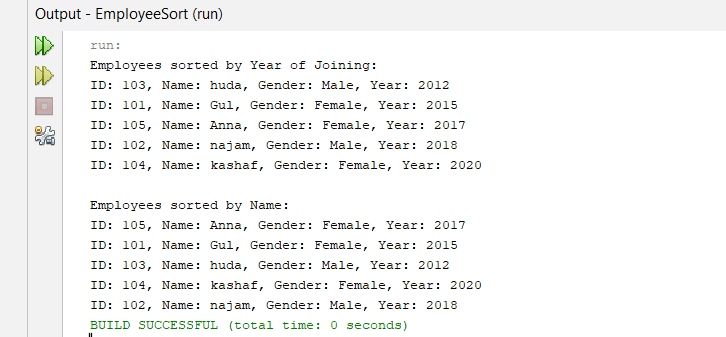
**Code:**

****

****

****

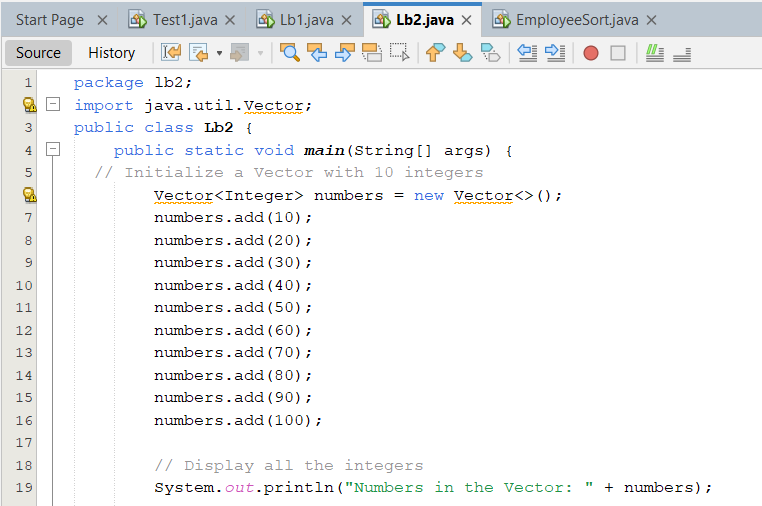
**Output:**

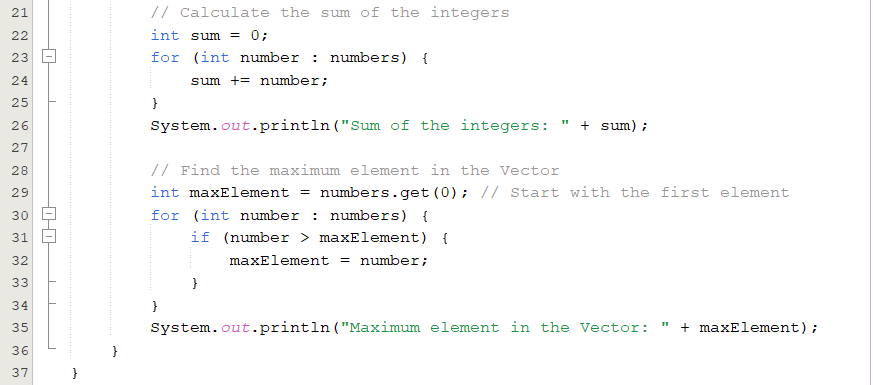
**

1. *Write a program that initializes Vector with 10 integers in it.*

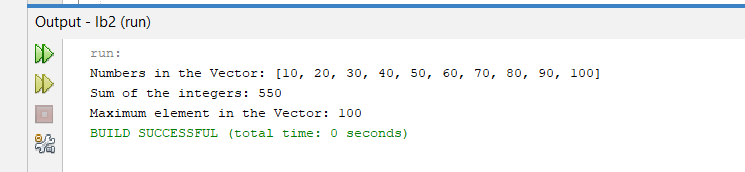
*• Display all the integers • Sum of these integers. • Find Maximum Element in Vector*

**Code:**

****

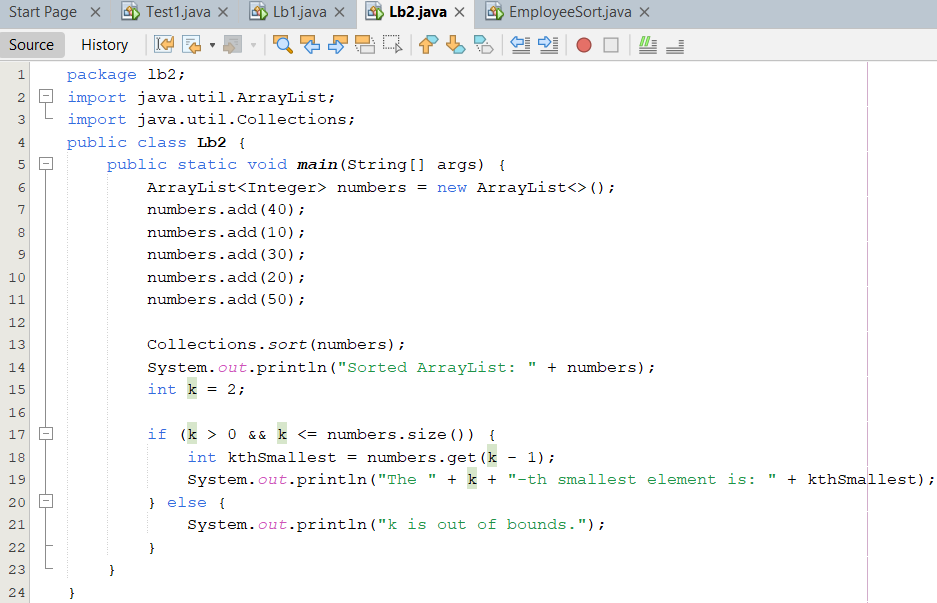
****

**Output:**

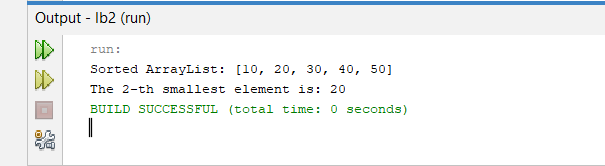
**

1. *Find the k-th smallest element in a sorted ArrayList*

**Code:**

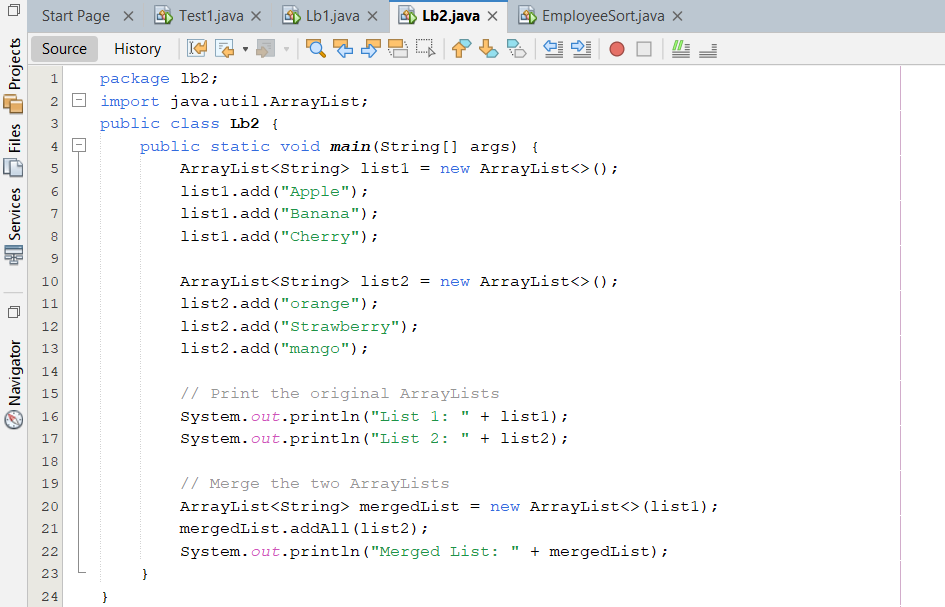
****

**Output:**

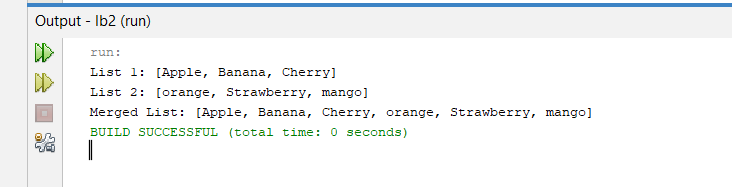
**

1. *Write a program to merge two ArrayLists into one.*

**Code:**

****

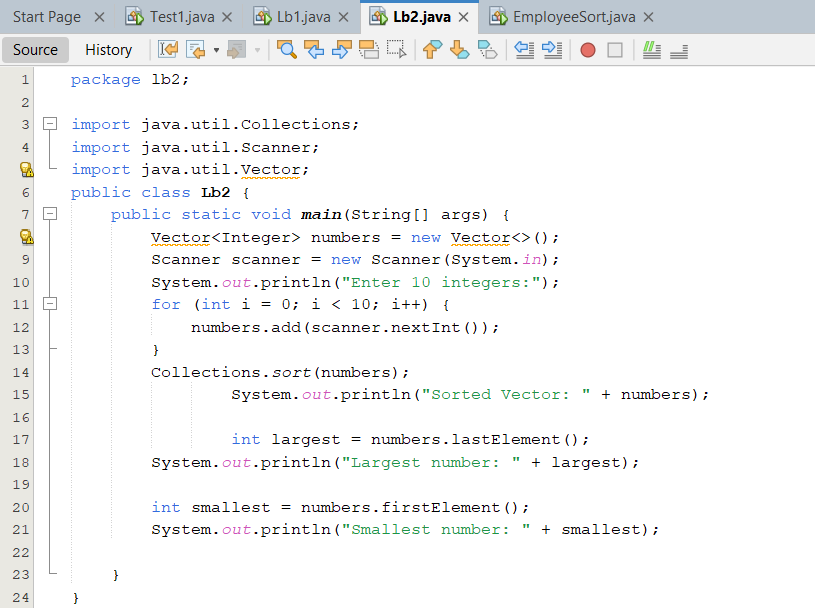
**Output:**

****

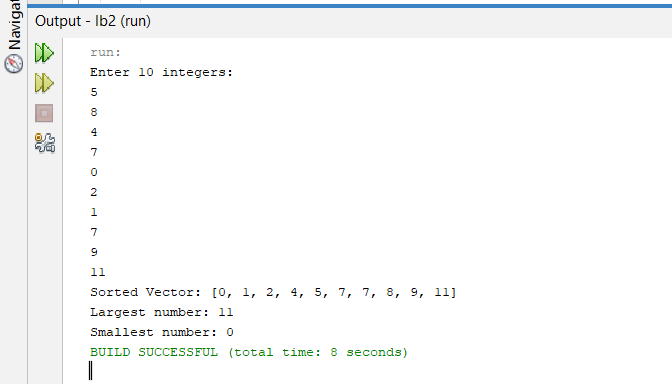
*HOME task*

1. *Create a Vector storing integer objects as an input. a. Sort the vector b. Display largest number c. Display smallest number*

**Code:**

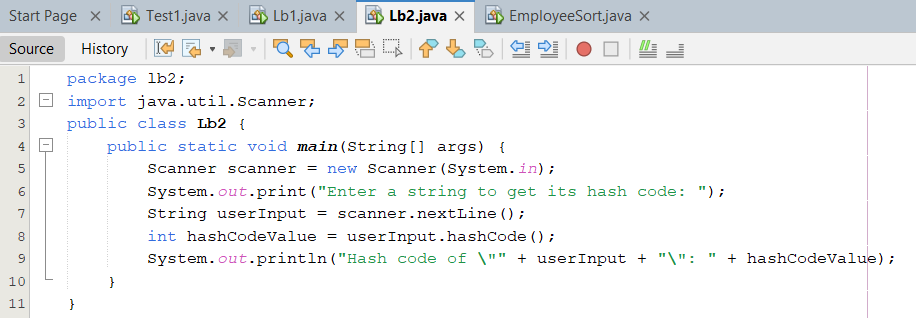
****

**Output:**

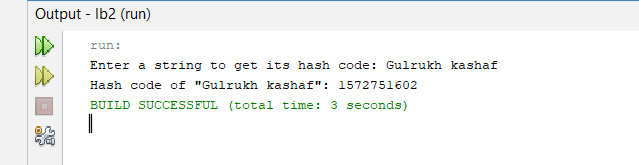
****

1. *Write a java program which takes user input and gives hashcode value of those inputs using hashCode () method.*

**Code:**

****

**Output:**

**

1. *Scenario based*

*Create a java project, suppose you work for a company that needs to manage a list of employees. Each employee has a unique combination of a name and an ID. Your goal is to ensure that you can track employees effectively and avoid duplicate entries in your system. Requirements*

1. *Employee Class: You need to create an Employee class that includes:*

*• name: The employee's name (String).*

*• id: The employee's unique identifier (int).*

*• Override the hashCode() and equals() methods to ensure that two employees are considered equal if they have the same name and id.*

1. *Employee Management: You will use a HashSet to store employee records. This will help you avoid duplicate entries.*
2. *Operations: Implement operations to:*

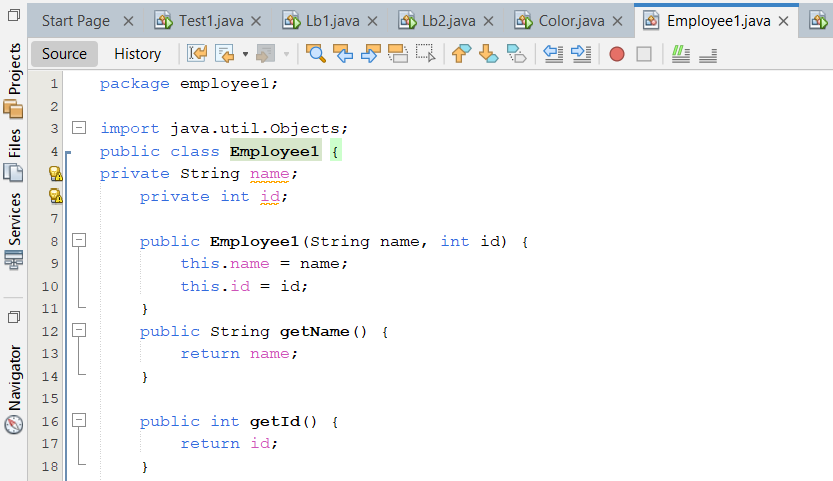
*• Add new employees to the record.*

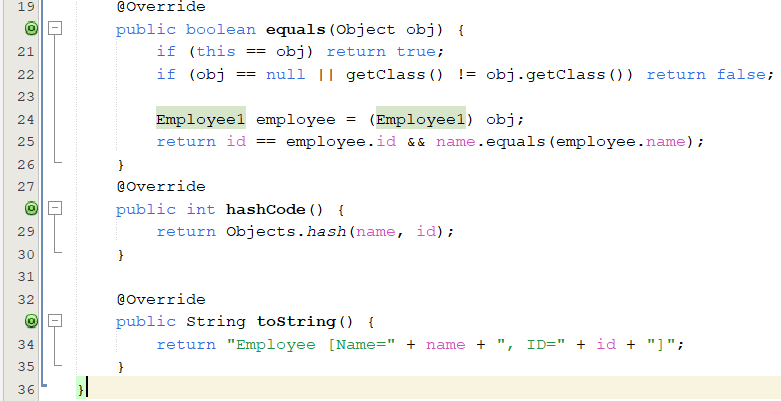
*• Check if an employee already exists in the records.*

*• Display all employees.*

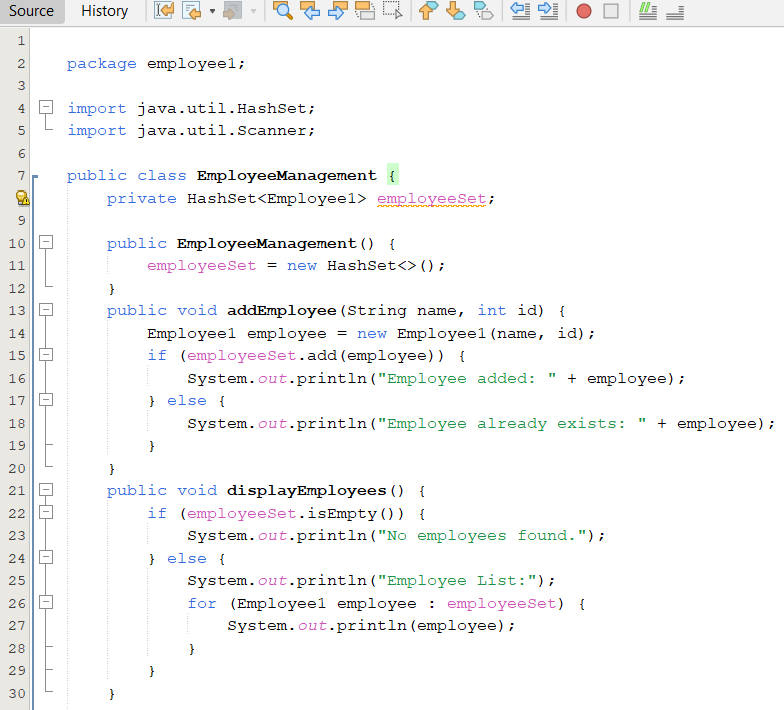
**Code:**

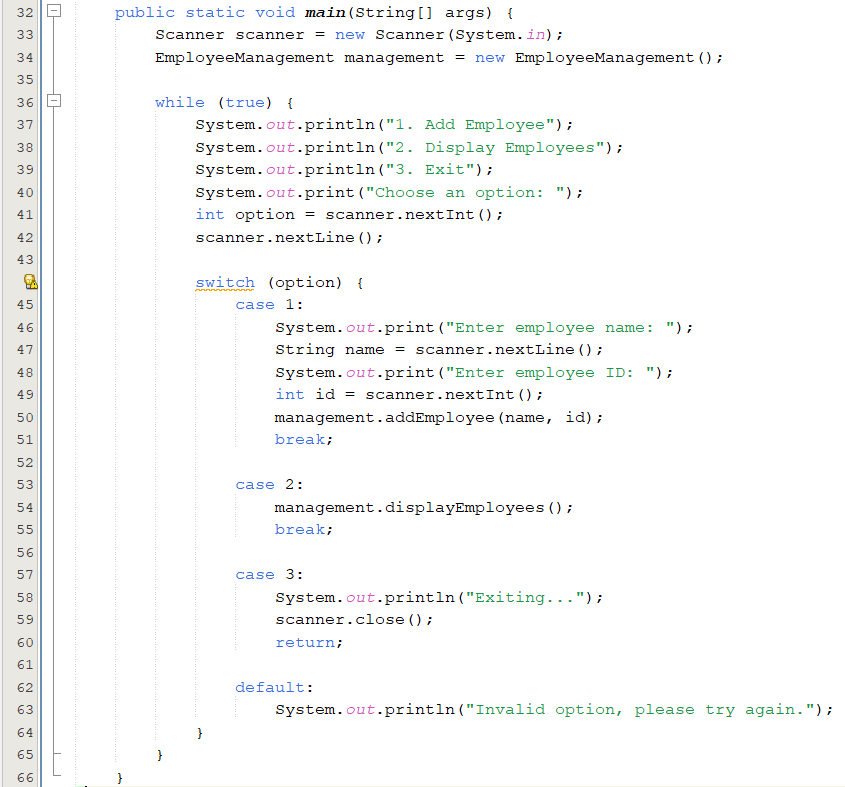
**Employee:**

****

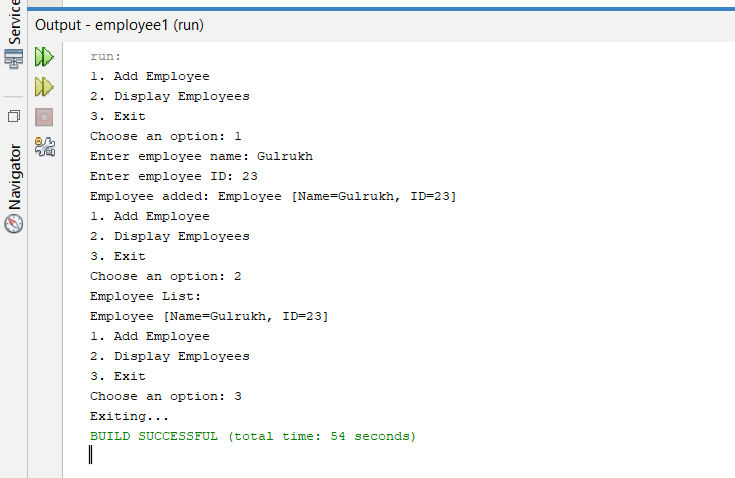
****

**Employee Management:**

****

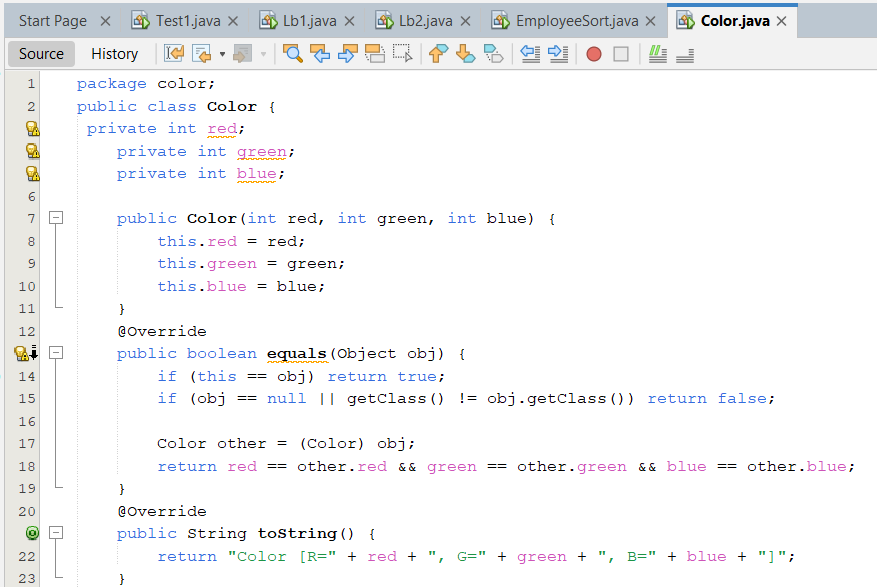
****

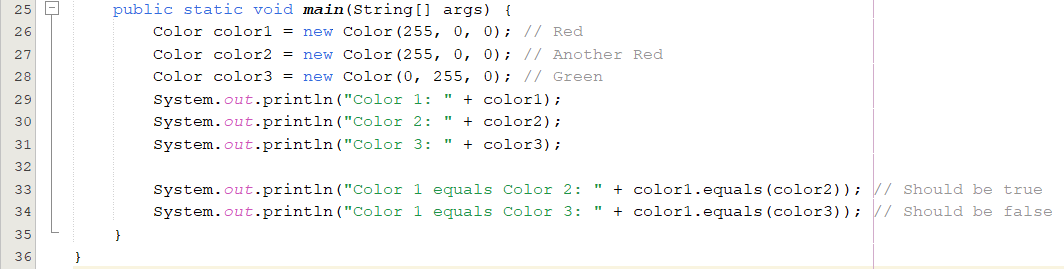
**Output:**

**

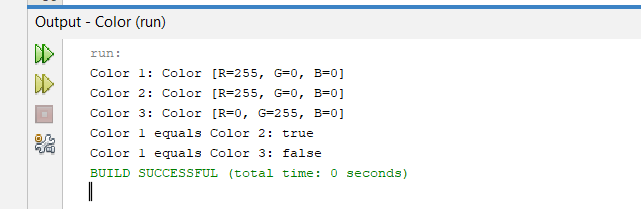
***4****.Create a Color class that has red, green, and blue values. Two colors are considered equal if their RGB values are the same*

**Code:**

****

****

**Output:**

**