Java Day: 2

write code to add items to integer, string **array**

int[] intArray= new int[3];

int[0]=1;

int[1]=2;

int[] intArray= {2,25};

String[] stringArray= new String[2];

String[0]=”Hello”;

String[1]=”World”;

String[] stringArray= {“Hello”,”World”};

write code to retrieve items from integer, string array

**import** java.util.Scanner;

**public** **class** ArrayDemo {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

**int**[] array = **new** **int**[5];

Scanner scanner = **new** Scanner(System.***in***);

System.***out***.println("Enter the values of the array");

**for**(**int** i=0;i<array.length;i++){

array[i] = scanner.nextInt();

}

System.***out***.println("Array elements are:");

**for**(**int** i=0;i<array.length;i++){

System.***out***.println(array[i]);

}

}

}

write code to add items to ArrayList collection

Ans :

List<String> list= new ArrayList<String>();

List.add(“Hello World”);

write code to retrieve items from arraylist (using for each loop\_

for(String str:list){

System.out.println(str);

}

write code to add items HashMap

Map<Integer,String> students = new HashMap<>();

students.put(1, "Hello");

students.put(2, "World");

write code to retrieve items HashMap

Map<Integer,String> students = new HashMap<>();

students.put(1, "Hello");

students.put(2, "World");

//get value by giving key value

System.out.println(students.get(1));

// get all keys

System.out.println(students.keySet());

Write code to add items to hashset

Write code to retrieve items to hasset

import java.util.HashSet;

import java.util.Set;

public class HashSetDemo {

public static void main(String[] args) {

// TODO Auto-generated method stub

//Set cannot take duplicate values

Set<String> set = new HashSet<>();

set.add("What");

set.add("are ");

set.add("you");

set.add("doing");

set.add("world");

System.out.println(set.size());

for(String s : set){

System.out.println(s);

}

set.remove("world");

System.out.println(names.size());

}

}

write code to connect to JDBC to get rows from employee table:

**public** **class** JDBCExample {

// JDBC driver name and database URL

**static** **final** String ***JDBC\_DRIVER*** = "com.mysql.jdbc.Driver";

**static** **final** String ***DB\_URL*** = "jdbc:mysql://localhost/STUDENTS";

// Database credentials

**static** **final** String ***USER*** = "username";

**static** **final** String ***PASS*** = "password";

**public** **static** **void** main(String[] args) {

Connection conn = **null**;

Statement stmt = **null**;

**try**{

Class.*forName*("com.mysql.jdbc.Driver");

System.***out***.println("Connecting to a selected database...");

conn = DriverManager.getConnection(***DB\_URL***, ***USER***, ***PASS***);

System.***out***.println("Connected database successfully...");

System.***out***.println("Creating statement...");

stmt = conn.createStatement();

String sql = "SELECT id, first, last, age FROM Employee";

ResultSet rs = stmt.executeQuery(sql);

**while**(rs.next()){

**int** id = rs.getInt("id");

**int** age = rs.getInt("age");

String first = rs.getString("first");

String last = rs.getString("last");

System.***out***.print("ID: " + id);

System.***out***.print(", Age: " + age);

System.***out***.print(", First: " + first);

System.***out***.println(", Last: " + last);

}

rs.close();

}**catch**(SQLException se){

se.printStackTrace();

}**catch**(Exception e){

e.printStackTrace();

}**finally**{

**try**{

**if**(stmt!=**null**)

conn.close();

}**catch**(SQLException se){

}

**try**{

**if**(conn!=**null**)

conn.close();

}**catch**(SQLException se){

se.printStackTrace();

} }

System.***out***.println("Goodbye!");

}

}

Write method to return list of rows code to loop throughs

create Employee class

public class Employee {

String fname;

String lname;

Employee(String fname,String lname){

this.fname = fname;

this.lname = lname;

}

Add employee class to list collection

create method that return list of employee collection

import java.util.ArrayList;

import java.util.List;

static List<Employee> emplist = new ArrayList<>();

public List<Employee> getEmpList(){

return emplist;

}

public static void main(String[] args) {

// TODO Auto-generated method stub

Employee e1 = new Employee("Nagarjuna","Reddy");

Employee e2 = new Employee("Mahesh","Reddy");

emplist.add(e1);

emplist.add(e2);

List<Employee> list = e1.getEmpList();

for(Employee e :list){

System.out.println(e.fname+" "+e.lname);

}

}

}

Difference between string, string buffer, string builder with example

String is Immutable and synchronized(Thread safe).

Ex: String str= “Hello”; hello is stored in constant String pool

String buffer is mutable and synchronized

StringBuffer str = new StringBuffer("Hello") ;

String Builder is mutable and not synchronized

StringBuilder str= new StringBuilder("Hello");