Scanner:

Scanner stdin = new Scanner (System.in);

Int x= stdin.nextInt();

Array :

*int*[] primes=new *int*[10];

String[] puppy = { “pika”, “mila”, “arlo”, “pikki” };

float[][] temperature=new float[10][365];

primes.**length;**

**Methods:**

public static void nPrintln (String message, int n) {

**}**

**Math.random();**

**Character.isLetter();**

**Character.isDigit();**

**String:**

**message.length();**

**String s3 = s1.concat(s2);**

**s1.substring(0, 11);**

**s1.equals(s2);**

**s1.compareTo(s2);**

**toLowerCase ();**

**toUpperCase ();**

**trim ();**

**s1.replace('a','e');**

**str.indexOf('W')**

**str.indexOf('W',5)**

**String.valueOf(value); # Integer.parseInt(strExp)**

**Files:**

* + - **inFile.next()**
    - **inFile.nextLine()**
    - **inFile.nextInt()**
    - **inFile.nextDouble()**
    - **inFile.hasNext()**
    - **outFile.print()**
    - **outFile.println()**
    - **outFile.printf()**
    - **inFile.close();**
    - **outFile.close();**

**Import java.io.\*;**

**Import java.util.\*;**

**Scanner inFile = new Scanner( new FileReader(“st.txt”)) ;**

**PrintWriter outfile = new PrintWriter(“output.txt”);**

**While (infile.hasnext())**

**{**

**Name=infile.next();**

**Outputfile.println(Name);**

**}**

**Infile.close();**

**Outfile.close();**

Array List declaration :

ArrayList<Type> name = new ArrayList<Type>();

Operations:

list.add(“Item1”);

add(index, value);

addAll(list)

addAll(index, list)

indexOf(value); 🡺 returns index

lastIndexOf(value);

int arrayListLength = list.Size();

String firstString = list.get(0);

list.Remove(0) // you can remove an object .remove ( you need just to specify the equals you defined)

removeAll(list)

retainAll(list)

subList(from, to)

IsEmpty()

Clear()

ToArray()

Set(Index, Object)

list.get(i).startsWith("B") 🡺 Boolean

list.contains("Nathaniel") 🡺 Boolean

containsAll(list);

equals(list);

**#Iterator**

Iterator itr=list.iterator();

while(itr.hasNext())

{

System.out.println(itr.next());

}

**#For each**

**for(String obj:list)**

**System.out.println(obj);**

Abstract Classes:

* + abstract class MyClass {
    - abstract void draw();
  + }

Interface:

public interface Drawable {  
 **void** draw();    
}

**class** Circle **implements** Drawable{

**public** **void** draw(){

System.out.println("drawing circle");}

}

#Default methods

default void newmethod()

{}

#Static

static void anothernewmethod()

{}

#CompareTo for sorting

#implements comparable <Student> in the class

Public int compareTo(Student st){

If (age==st.age){return 0}

Else (if age>st.age){return 1}

Else return -1;

}

Arrays.sort(S)

For arraylist :

Collections.sort(list);

#equals override

public boolean equals(Object obj) {

if(!(obj instanceof Software))

return false;

boolean isEqual;

isEqual = (this.id == ((Software) obj).id);

return isEqual;

}

#Database:

**import java.sql.Connection;  
import java.sql.DriverManager;  
import java.sql.SQLException;**

**import java.sql.Statement;**

**import java.sql.ResultSet;**

**try {**

**Connection con = DriverManager.getConnection(** host, username, password **);**

**Statement stmt = con.createStatement( );**

**String SQL = "SELECT \* FROM Workers";**

**ResultSet rs = stmt.executeQuery( SQL );**

**int id\_col = rs.getInt("ID");**

**}  
catch ( SQLException err ) {  
System.out.println( err.getMessage( ) );  
}**

**Prepared statement :**

**SQL="DELETE FROM Users WHERE ID = ?";**

**SQL="UPDATE Users SET(?,?,?,?) WHERE ID=Id";**

**SQL="INSERT INTO Users VALUES(?,?,?,?)";**

**pstmt = con.prepareStatement(SQL);**

**pstmt.setInt(1, Integer.parseInt(ID.getText()));**

**pstmt.setString(2, Fname.getText());**

**pstmt.setString(3, Lname.getText());**

**pstmt.setInt(4, Integer.parseInt(Age.getText()));**

**count = pstmt.executeUpdate();**

**GUI**

**#How to keep a field enabled when radio button is selected**

**price.setEnabled(false);**

**price.setEnabled(true);**

#New frame:

ListF Frame = new ListF() ;

Frame.setVisible(true);

#Fill table

private void fillJtable() {

try

{

DefaultTableModel model = new DefaultTableModel();

Object[] columnsName = new Object[4];

columnsName[0] = "ID";

columnsName[1] = "First Name";

columnsName[2] = "Last Name";

columnsName[3] = "Age";

model.setColumnIdentifiers(columnsName);

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

Connection con = DriverManager.getConnection("jdbc:derby://localhost:1527/DBuserscontaining","Khouloud","123");

Statement stmt = con.createStatement( );

String SQL = "SELECT \* FROM Users";

ResultSet rs = stmt.executeQuery( SQL );

while(rs.next())

{

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

String first\_name = rs.getString("FName");

String last\_name = rs.getString("LName");

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

Object row [] = {id, first\_name, last\_name, age};

model.addRow(row);

}

jTable1.setModel(model);

}

catch(ClassNotFoundException | SQLException x)

{

System.out.println("Exception :" + x);

}

}

#COMBOBOX

private void getData(){

try {

Connection con = DriverManager.getConnection("jdbc:derby://localhost:1527/EmployeeList","employee","123");

Statement stm = con.createStatement();

String SQL = "SELECT \* FROM Department";

ResultSet rs = stm.executeQuery( SQL );

while (rs.next()){

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

combox.addItem(name);

}

con.close();

} catch (SQLException ex) {

System.out.println(ex);

}}

Retrieve:

Combo.getSelectedItem().toString();