**Program Eleven Part One:**

//summary: uses the ArrayList class to take in as many numbers as the user wants and then finds the largest

//name: Jenna Wolf

//class: Fundamentals of Programming, CS155 - 01

//instructor: Dr. Art Kazmierczak

//date: 10/30/2023

import java.util.Scanner; //lets the user make inputs

import java.util.ArrayList; //lets the ArrayList class be used

public class Main

{

public static void main(String[] args)

{

Scanner input = new Scanner(System.in); //names the input

ArrayList<Integer> list = new ArrayList<Integer>(); //holds the list data

int num; //holds the num data

//asks the user for a sequence of numbers

System.out.println("Please enter a sequence of numbers ending with 0: ");

num = input.nextInt(); //takes in the users input

//while num does not equal 0, the loop continues

while(num != 0)

{

list.add(num); //adds num to list

num = input.nextInt(); //takes in the next number

}

//outputs the max number in the list

System.out.println("The max number in the array is " + max(list));

}

//finds the largest number

public static Integer max(ArrayList<Integer> list)

{

int max; //holds the max data

//checks to see if list is empty and returns null

if(list.isEmpty())

return null;

else

{

max = list.get(0); //sets max to the first element on the list

//for loop that goes until i is bigger than the list size

for(int i = 0; i < list.size(); i++)

{

//checks if max is less than the current list number

if(max < list.get(i))

max = list.get(i); //max is set to the current list number

}

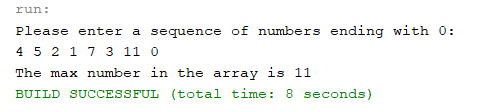
return max; //returns max

}

}

}

Output:



**Program Eleven Part Two:**

//summary: This program uses the ArrayList to store data of different data types and output each in string form

//name: Jenna Wolf

//class: Fundamentals of Programming, CS155 - 01

//instructor: Dr. Art Kazmierczak

//date: 10/30/2023

import java.util.ArrayList; //allows ArrayList to be used

import java.util.Date; //allows Date to be used

public class Main

{

public static void main(String[] args)

{

ArrayList<Object> list = new ArrayList<Object>(); //holds the list data

list.add(new loan(0.12, 5, 5000)); //adds a loan to the list

list.add(new Date()); //adds a Date to the list

list.add(new String("I'm a string!")); //adds a String to the list

list.add(new circle(11)); //adds a circle to the list

//outputs all data in the List

System.out.println((list.get(0)).toString());

System.out.println((list.get(1)).toString());

System.out.println((list.get(2)).toString());

System.out.println((list.get(3)).toString());

}

}

**Loan**

import java.util.Date; //allows the Date data type to be used

public class loan {

private double annualInterestRate; //holds the annualInterestRate data

private int numberOfYears; //holds the numberOfYears data

private double loanAmount; //holds the loanAmount data

private Date loanDate = new Date(); //holds the loanDate data

//no args constructor

loan() {

}

//constructor that takes in 2 doubles and an int

loan(double temp1, int temp2, double temp3) {

annualInterestRate = temp1;

numberOfYears = temp2;

loanAmount = temp3;

loanDate = new Date();

}

//returns the annualInterestRate data

public double getAnnualInterestRate()

{

return annualInterestRate;

}

//returns the numberOfYears data

public int getNumberOfYears()

{

return numberOfYears;

}

//returns the loanAmount data

public double getLoanAmount()

{

return loanAmount;

}

//returns the loanDate data

public Date getLoanDate()

{

return loanDate;

}

//sets the annualInterestRate to the double sent over

public void setAnnualInterestRate(double temp)

{

annualInterestRate = temp;

}

//sets the numberOfYears to the int sent over

public void setNumberOfYears(int temp)

{

numberOfYears = temp;

}

//sets the loanAmount to the double sent over

public void setLoanAmount(double temp)

{

loanAmount = temp;

}

//caculates the monthly payment

public double getMonthlyPayment()

{

double monthlyInterest = annualInterestRate / 1200; //finds the monthly interest rate

//caculates and returns the monthly payment amount

return loanAmount \* monthlyInterest / (1 - (1 / Math.pow(1 + monthlyInterest, numberOfYears \* 12)));

}

//caculates the total amount

public double getTotalAmount()

{

//caculates and returns the total amount

return getMonthlyPayment() \* numberOfYears \* 12;

}

//uses the loan data to return a string

public String toString()

{

return String.format("Total Amount of loan: %.2f", getTotalAmount());

}

}

**Circle**

ublic class circle

{

private double radius = 1; //holds the radius data

//no args constructor

circle(){

}

//constructor that takes in a double

circle(double temp){

radius = temp;

}

//returns the radius data

public double getRadius()

{

return radius;

}

//sets radius to the double sent over

public void setRadius(double temp)

{

radius = temp;

}

//caculates the area

public double getArea()

{

//caculates and returns the radius

return radius \* radius \* Math.PI;

}

//caculates the perimeter

public double getPerimeter()

{

//caculates and returns the perimeter

return 2 \* radius \* Math.PI;

}

//uses the circle data to return a string

public String toString()

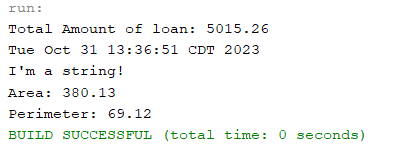
{

return String.format("Area: %.2f", getArea()) + String.format("\nPerimeter: %.2f", getPerimeter());

}

}

Output:



**Program Eleven Part Three:**

//summary: makes a class named MyStack that serves as a extension to ArrayList. Then takes in 5 strings from the

//user and returns them in backwards order

//name: Jenna Wolf

//class: Fundamentals of Programming, CS155 - 01

//instructor: Dr. Art Kazmierczak

//date: 10/30/2023

import java.util.Scanner; //includes the Scanner class

public class Main

{

public static void main(String[] args)

{

Scanner input = new Scanner(System.in); //lets the user makes inputs

MyStack stack = new MyStack(); //holds the stack data

//takes in 5 strings from the user

System.out.println("Enter 5 strings: ");

for(int i = 0; i < 5; i++)

{

String temp = input.nextLine();

stack.push(temp);

}

//outputs the strings the user entered

for(int i = 0; i < 5; i++)

System.out.println(stack.pop());

}

}

**MyStack**

import java.util.ArrayList; //allows the ArrayList class to be used

public class MyStack {

private ArrayList<Object> list = new ArrayList<>(); //holds the list data

//no args constructor

public MyStack() {

list = new ArrayList<>();

}

//checks to see if list is empty

public boolean isEmpty()

{

return list.isEmpty();

}

//gets the size of list

public int getSize()

{

return list.size();

}

//sees what the last object in peek is

public Object peek()

{

return list.get(getSize() - 1);

}

//returns and removes the last item in list

public Object pop()

{

Object temp = list.get(getSize() - 1); //sets temp to the last item in list

list.remove(getSize() - 1); //removes the last item in list

return temp; //returns temp

}

//adds an item to list

public void push(Object temp)

{

list.add(temp);

}

//sends back a string containg all items in list

public String toString()

{

return "Stack: " + list.toString();

}

}

Output:

