Bahria University,

Karachi Campus

A logo with text on it

Description automatically generated

LAB EXPERIMENT NO.

06

LIST OF TASKS

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
| **01** | Implement the Fully Functional Calculator in Android Studio As per given  Picture: |
|  |  |
|  |  |

Submitted On:

29-03-2024

(Date: DD/MM/YYYY)

**Task No. 01**: Implement the Fully Functional Calculator in Android Studio As per given

Picture:

**Solution:**

**Activity\_Main.java**

package com.example.scientificcalculator;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

Button b1,b2,b3,b4,b5,b6,b7,b8,b9,b0,bac,bc,bp,bsqrt,bplus,bminus,bdiv,bmod,bequal,bdot,bbrac1,bbrac2,bsin,bcos,btan,bsquare,bpi,bfact,blog,bln,binv;

TextView tvmain,tvsec;

int sum = 0;

String pi = "3.14159265";

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

b1 = findViewById(R.id.b1);

b2 = findViewById(R.id.b2);

b3 = findViewById(R.id.b3);

b4 = findViewById(R.id.b4);

b5 = findViewById(R.id.b5);

b6 = findViewById(R.id.b6);

b7 = findViewById(R.id.b7);

b8 = findViewById(R.id.b8);

b9 = findViewById(R.id.b9);

b0 = findViewById(R.id.b0);

bac = findViewById(R.id.bac);

bc = findViewById(R.id.bc);

bplus = findViewById(R.id.bplus);

bminus = findViewById(R.id.bminus);

bdiv = findViewById(R.id.bdiv);

bmod = findViewById(R.id.bmod);

bequal = findViewById(R.id.bequal);

bdot = findViewById(R.id.bdot);

bsqrt = findViewById(R.id.bsqrt);

bfact = findViewById(R.id.bfact);

bsquare = findViewById(R.id.bsquare);

bsin = findViewById(R.id.bsin);

bcos = findViewById(R.id.bcos);

btan = findViewById(R.id.btan);

bpi = findViewById(R.id.bpi);

bbrac1 = findViewById(R.id.bbrac1);

bbrac2 = findViewById(R.id.bbrac2);

blog = findViewById(R.id.blog);

bln = findViewById(R.id.bln);

binv = findViewById(R.id.binv);

tvmain = findViewById(R.id.tvmain);

tvsec = findViewById(R.id.tvsec);

//onclick listeners

b1.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

tvmain.setText(val+b1.getText().toString());

}

});

b2.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

tvmain.setText(val+b2.getText().toString());

}

});

b3.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

tvmain.setText(val+b3.getText().toString());

}

});

b4.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

tvmain.setText(val+b4.getText().toString());

}

});

b5.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

tvmain.setText(val+b5.getText().toString());

}

});

b6.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

tvmain.setText(val+b6.getText().toString());

}

});

b7.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

tvmain.setText(val+b7.getText().toString());

}

});

b8.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

tvmain.setText(val+b8.getText().toString());

}

});

b9.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

tvmain.setText(val+b9.getText().toString());

}

});

b0.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

tvmain.setText(val+b0.getText().toString());

}

});

bdot.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

if (!val.contains("."))

{

tvmain.setText(val+bdot.getText().toString());

}

}

});

bplus.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

if (!val.equals(""))

{

tvmain.setText(val+bplus.getText().toString());

}

}

});

bdiv.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

if (!val.equals(""))

{

tvmain.setText(val+bdiv.getText().toString());

}

}

});

bminus.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

char last = val.charAt(val.length() -1);

if (last!='-')

{

tvmain.setText(val+bminus.getText().toString());

}

}

});

bmod.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

char last = val.charAt(val.length() -1);

if (!val.equals(""))

{

tvmain.setText(val+bmod.getText().toString());

}

}

});

bsqrt.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

double r = Math.sqrt(Double.parseDouble(val));

String result = String.valueOf(r);

tvmain.setText(result);

}

});

bequal.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

String replacedString = val.replace('÷','/').replace('×', '\*');

double result = eval(replacedString);

String r = String.valueOf(result);

tvmain.setText(r);

tvsec.setText(val);

}

});

bac.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

tvmain.setText("");

tvsec.setText("");

}

});

bc.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String val = tvmain.getText().toString();

if (!val.equals(""))

{

val = val.substring(0, val.length() - 1);

tvmain.setText(val);

}

}

});

bbrac1.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

tvmain.setText(tvmain.getText()+"(");

}

});

bbrac2.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

tvmain.setText(tvmain.getText()+")");

}

});

bpi.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

tvmain.setText(tvmain.getText()+pi);

tvsec.setText(bpi.getText());

//hold

}

});

bsin.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

tvmain.setText(tvmain.getText()+"sin");

//hold

}

});

bcos.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

tvmain.setText(tvmain.getText()+"cos");

//hold

}

});

btan.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

tvmain.setText(tvmain.getText()+"tan");

//hold

}

});

bsquare.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

double d = Double.parseDouble(tvmain.getText().toString());

double square = d\*d;

tvmain.setText(String.valueOf(square));

tvsec.setText(d+"²");

}

});

bfact.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

int val = Integer.parseInt(tvmain.getText().toString());

int fact = factorial(val);

tvmain.setText(String.valueOf(fact));

tvsec.setText(val+"!");

}

});

binv.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

tvmain.setText(tvmain.getText().toString()+"^"+"(-1)");

}

});

bln.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

tvmain.setText(tvmain.getText().toString()+"ln");

}

});

blog.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

tvmain.setText(tvmain.getText().toString()+"log");

}

});

}

//factorial

int factorial(int n)

{

// find factorial

return (n == 1 || n == 0) ? 1 : n \* factorial(n - 1);

}

//evaluation

public static double eval(final String str) {

return new Object() {

int pos = -1, ch;

void nextChar() {

ch = (++pos < str.length()) ? str.charAt(pos) : -1;

}

boolean eat(int charToEat) {

while (ch == ' ') nextChar();

if (ch == charToEat) {

nextChar();

return true;

}

return false;

}

double parse() {

nextChar();

double x = parseExpression();

if (pos < str.length()) throw new RuntimeException("Unexpected: " + (char)ch);

return x;

}

// Grammar:

// expression = term | expression `+` term | expression `-` term

// term = factor | term `\*` factor | term `/` factor

// factor = `+` factor | `-` factor | `(` expression `)`

// | number | functionName factor | factor `^` factor

double parseExpression() {

double x = parseTerm();

for (;;) {

if (eat('+')) x += parseTerm(); // addition

else if (eat('-')) x -= parseTerm(); // subtraction

else return x;

}

}

double parseTerm() {

double x = parseFactor();

for (;;) {

if (eat('\*')) x \*= parseFactor(); // multiplication

else if (eat('/')) x /= parseFactor(); // division

else return x;

}

}

double parseFactor() {

if (eat('+')) return parseFactor(); // unary plus

if (eat('-')) return -parseFactor(); // unary minus

double x;

int startPos = this.pos;

if (eat('(')) { // parentheses

x = parseExpression();

eat(')');

} else if ((ch >= '0' && ch <= '9') || ch == '.') { // numbers

while ((ch >= '0' && ch <= '9') || ch == '.') nextChar();

x = Double.parseDouble(str.substring(startPos, this.pos));

} else if (ch >= 'a' && ch <= 'z') { // functions

while (ch >= 'a' && ch <= 'z') nextChar();

String func = str.substring(startPos, this.pos);

x = parseFactor();

if (func.equals("sqrt")) x = Math.sqrt(x);

else if (func.equals("sin")) x = Math.sin(Math.toRadians(x));

else if (func.equals("cos")) x = Math.cos(Math.toRadians(x));

else if (func.equals("tan")) x = Math.tan(Math.toRadians(x));

else if (func.equals("log")) x = Math.log10(x);

else if (func.equals("ln")) x = Math.log(x);

else throw new RuntimeException("Unknown function: " + func);

} else {

throw new RuntimeException("Unexpected: " + (char)ch);

}

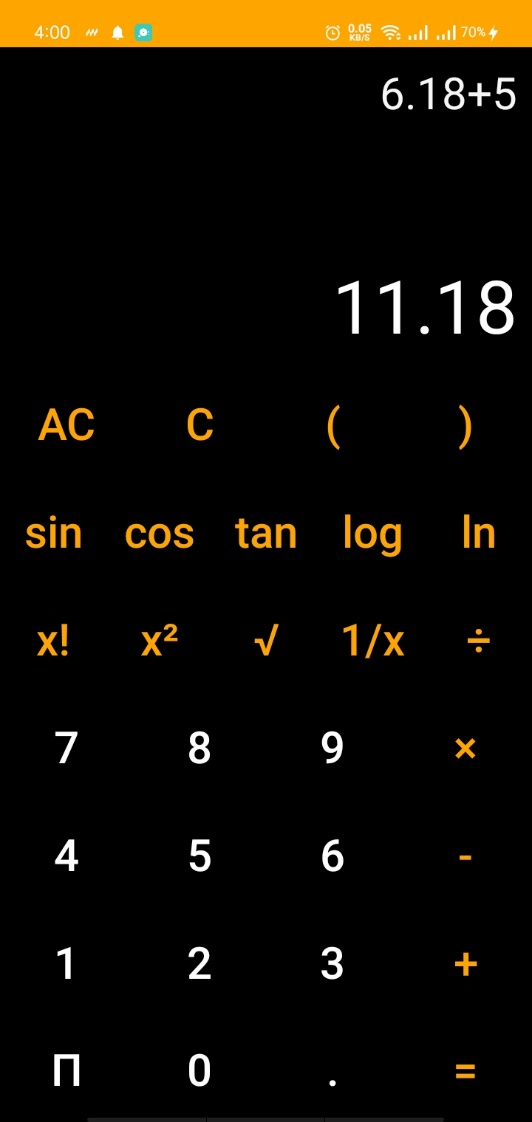
if (eat('^')) x = Math.pow(x, parseFactor()); // exponentiation

return x;

}

}.parse();

}

}

A screenshot of a calculator

Description automatically generated**Output:**

**A screenshot of a calculator

Description automatically generated**