**import** android.support.v7.app.AppCompatActivity;

**import** android.os.Bundle;

**import** android.util.Log;

**import** android.view.View;

**import** android.widget.Button;

**import** android.widget.EditText;

**import** java.util.Objects;

**public class** MainActivity **extends** AppCompatActivity {

*//For debugging purposes*

**public static final** String ***TAG*** = MainActivity.**class**.getSimpleName();

*//Initialise general variables used for calculator*

String **current\_workspace**;

String **current\_equation**;

String **current\_operation** = **"null"**;

String **past\_operation**;

*//Initialise boolean control variables*

**boolean equals\_last** = **false**;

**boolean waiting** = **false**;

**boolean crush520** = **false**;

**boolean valid** = **false**;

*//Initialise variables for calculation*

String **temp**;

**float number1**;

**float number2**;

*//EditText variables to display string/numbers*

EditText **workspace**;

EditText **equation**;

*//Every Activity made is started through a sequence of method calls.*

*//onCreate() is the first of these calls.*

@Override

**protected void** onCreate(Bundle savedInstanceState) {

**super**.onCreate(savedInstanceState);

setContentView(R.layout.***activity\_main***);

**workspace** = (EditText) findViewById(R.id.***workspace***);

**equation** = (EditText) findViewById(R.id.***equation***);

}

*//Function for button 0 (includes Log.d which is used for debugging purposes)*

**public void** button\_0(View view) {

Log.*d*(***TAG***,**"Button 0 is pressed"**);

number\_button(**"0"**);

}

*//Function for buttons 1 - 9, passes a string of the number*

*//into the function number\_button*

**public void** button\_1(View view) {

number\_button(**"1"**);

}

**public void** button\_2(View view) {

number\_button(**"2"**);

}

**public void** button\_3(View view) {

number\_button(**"3"**);

}

**public void** button\_4(View view) {

number\_button(**"4"**);

}

**public void** button\_5(View view) {

number\_button(**"5"**);

}

**public void** button\_6(View view) {

number\_button(**"6"**);

}

**public void** button\_7(View view) {

number\_button(**"7"**);

}

**public void** button\_8(View view) {

number\_button(**"8"**);

}

**public void** button\_9(View view) {

number\_button(**"9"**);

}

*//Function input String x, x of the button.*

**public void** number\_button(String x){

*//This if would run if the previous state has already been calculated*

*//If equals\_last is false it would mean that number is already present*

*//within the current workspace.*

*//Obtain the current string from both current\_workspace and current\_equation*

*//and display them again.*

*//crush520 is for the special function*

**if** (**equals\_last** == **false** && **crush520** == **false**) {

**current\_workspace** = **workspace**.getText().toString();

**current\_workspace** += x;

**workspace**.setText(**current\_workspace**);

**current\_equation** = **equation**.getText().toString();

**current\_equation** += x;

**equation**.setText(**current\_equation**);

}

*//Else it would mean that the last function called is the "equals" button.*

*//clear the current workspace by setting the current text to input x*

*//set equals\_last to be false*

*//crush520 is for the special function*

**else** {

**workspace**.setText(x);

**equation**.setText(x);

**current\_equation** = x;

**crush520** = **false**;

**equals\_last** = **false**;

}

**valid** = **true**;

}

*//STOP ONCE TO DEMO*

*//Each operator calculates the previous operation first before proceeding*

*//9 + 9 + 9*

*//The first operator is "past operation"*

*//The second operator is "current operation" which would be saved to "past operation"*

*//When the second operator is keyed in, the first 2 digits would be*

**public void** button\_operator(String operator, String sign) {

*// valid only == true if the previous input is a number and not an operator*

*// preventing occurences of ++, +=*

**if** (**valid** == **true**) {

*//saves the current number in the workspace into the variable temp*

**temp** = **workspace**.getText().toString();

*//if there is no previous operation, i.e. first operation*

**if** (**current\_operation** == **"null"**){

**if** (!**temp**.equals(**""**)){

*//save number stored into temp into number1*

**number1** = Float.*parseFloat*(**temp**);

*//clear the workspace*

**workspace**.setText(**""**);

*//set input as current\_operation*

**current\_operation** = operator;

**current\_equation** = **equation**.getText().toString();

**current\_equation** += sign;

*//add sign to current\_equation*

**equation**.setText(**current\_equation**);

}

**else** {

*//preventing first input to be an operator*

**current\_operation** = **"null"**;

}

}

**if** (**past\_operation** == **"add"**) {

*//if past\_operation is add carry out add function*

*//convert from string to number, and assigning it to variable number 2*

**number2** = Float.*parseFloat*(**temp**);

*//short form for number1 = number1 + number2*

**number1** += **number2**;

**current\_operation** = operator;

**workspace**.setText(**""**);

*//adding the sign to the current equation*

**current\_equation** = **equation**.getText().toString();

**current\_equation** += sign;

**equation**.setText(**current\_equation**);

}

**else if** (**past\_operation** == **"minus"**) {

**number2** = Float.*parseFloat*(**temp**);

**number1** -= **number2**;

**current\_operation** = operator;

**workspace**.setText(**""**);

**current\_equation** = **equation**.getText().toString();

**current\_equation** += sign;

**equation**.setText(**current\_equation**);

}

**else if** (**past\_operation** == **"multiply"**){

**number2** = Float.*parseFloat*(**temp**);

**number1** = **number1** \* **number2**;

**current\_operation** = operator;

**workspace**.setText(**""**);

**current\_equation** = **equation**.getText().toString();

**current\_equation** += sign;

**equation**.setText(**current\_equation**);

}

**else if** (**past\_operation** == **"divide"**){

**number2** = Float.*parseFloat*(**temp**);

*//check for math error, divisible by 0*

**if** (**number2** == 0) {

**current\_workspace** = **""**;

**current\_operation** = **"null"**;

**equals\_last** = **false**;

**temp** = **""**;

**number1** = 0;

**number2** = 0;

**workspace**.setText(**"Math Error"**);

**equation**.setText(**""**);

}

**else** {

**number1** /= **number2**;

**current\_operation** = operator;

**workspace**.setText(**""**);

**current\_equation** = **equation**.getText().toString();

**current\_equation** += sign;

**equation**.setText(**current\_equation**);

}

}

**if** (!**current\_operation**.equals(**"null"**)) {

**past\_operation** = operator;

}

}

**valid** = **false**;

}

**public void** button\_add(View view) {

button\_operator(**"add"**,**"+"**);

}

**public void** button\_minus(View view) {

button\_operator(**"minus"**,**"-"**);

}

**public void** button\_multiply(View view) {

button\_operator(**"multiply"**, **"X"**);

}

**public void** button\_divide(View view) {

button\_operator(**"divide"**, **"/"**);

}

*//STOP TO SHOW OPERATOR*

*//Almost similar to the button\_operator function*

*//Adds the special function feature*

*//Automatically clears the workspace on next input based on the boolean equals\_last*

**public void** button\_equals(View view) {

**if** (**valid** == **true**) {

**temp** = **workspace**.getText().toString();

**if** (**current\_operation** == **"null"**){

*//check for the conditions listed for special feature*

**if** (**temp**.equals(**"160297"**)) {

**workspace**.setText(**"Happy Birthday!"**);

**crush520** = **true**;

**current\_workspace** = **""**;

**current\_operation** = **"null"**;

**equals\_last** = **false**;

**past\_operation** = **""**;

**temp** = **""**;

**number1** = 0;

**number2** = 0;

**equation**.setText(**""**);

}

**else if** (**temp**.equals(**"1402"**)) {

**workspace**.setText(**"Happy Valentines!"**);

**crush520** = **true**;

**current\_workspace** = **""**;

**current\_operation** = **"null"**;

**equals\_last** = **false**;

**past\_operation** = **""**;

**temp** = **""**;

**number1** = 0;

**number2** = 0;

**equation**.setText(**""**);

}

**else** {

*//if there are no operators selected*

*//number is keyed and equals is pressed*

**workspace**.setText(**temp**);

}

}

**if** (**past\_operation** == **"add"**){

*//obtain current number and save it to number2*

**number2** = Float.*parseFloat*(**temp**);

**number1** += **number2**;

**temp** = String.*valueOf*(**number1**);

*//Displaying the value back to workspace*

**workspace**.setText(**temp**);

*//Resetting the variables for next usage.*

**current\_equation** = **temp**;

**current\_workspace** = **""**;

**current\_operation** = **"null"**;

**equals\_last** = **true**;

**past\_operation** = **""**;

**temp** = **""**;

**number1** = 0;

**number2** = 0;

**equation**.setText(**""**);

}

**else if** (**past\_operation** == **"minus"**){

**number2** = Float.*parseFloat*(**temp**);

**number1** -= **number2**;

**temp** = String.*valueOf*(**number1**);

**workspace**.setText(**temp**);

**current\_equation** = **temp**;

**current\_operation** = **"null"**;

**current\_workspace** = **""**;

**equals\_last** = **true**;

**past\_operation** = **""**;

**temp** = **""**;

**number1** = 0;

**number2** = 0;

**equation**.setText(**""**);

}

**else if** (**past\_operation** == **"multiply"**){

**number2** = Float.*parseFloat*(**temp**);

**number1** \*= **number2**;

**temp** = String.*valueOf*(**number1**);

**workspace**.setText(**temp**);

**current\_equation** = **temp**;

**current\_operation** = **"null"**;

**current\_workspace** = **""**;

**equals\_last** = **true**;

**past\_operation** = **""**;

**temp** = **""**;

**number1** = 0;

**number2** = 0;

**equation**.setText(**""**);

}

**else if** (**past\_operation** == **"divide"**){

**number2** = Float.*parseFloat*(**temp**);

**if** (**number2** == 0) {

**current\_workspace** = **""**;

**current\_operation** = **"null"**;

**equals\_last** = **true**;

**temp** = **""**;

**number1** = 0;

**number2** = 0;

**workspace**.setText(**"Math Error"**);

**equation**.setText(**""**);

}

**else** {

**number1** /= **number2**;

**temp** = String.*valueOf*(**number1**);

**workspace**.setText(**temp**);

**current\_equation** = **temp**;

**current\_operation** = **"null"**;

**current\_workspace** = **""**;

**equals\_last** = **true**;

**past\_operation** = **""**;

**temp** = **""**;

**number1** = 0;

**number2** = 0;

**equation**.setText(**""**);

}

}

**past\_operation** = **""**;

}

**valid** = **false**;

}

*//STOP TO SHOW HAPPY VALENTINES*

**public void** button\_clear(View view) {

clear();

}

**public void** clear() {

*//Resetting the variables*

**current\_workspace** = **""**;

**current\_operation** = **"null"**;

**equals\_last** = **true**;

**valid** = **false**;

**past\_operation** = **""**;

**temp** = **""**;

**number1** = 0;

**number2** = 0;

*//Clearing the workspace and equation*

**workspace**.setText(**""**);

**equation**.setText(**""**);

}

}