Ondokuz Mayıs Üniversitesi Mobil Programlama Ödevi

Github Link

(https://github.com/bgebes/universityHomeworks-flutter/tree/master/calculator)

Ekran Görselleri

```
bgebes@dev:~/Development/universityHomeworks-flutter/calc...
                                                                 Q
  calculator git:(master) X dart main.dart
Type an operation: 2
You typed in wrong format! Try again! Format: {number1} {operator} {number2}
Type an operation: 2 5
You typed in wrong format! Try again! Format: {number1} {operator} {number2}
Type an operation: 2+5
You typed in wrong format! Try again! Format: {number1} {operator} {number2}
Type an operation: 2 + 5
2.0 + 5.0 = 7.0
Type an operation: 2 5 6
You typed unavailable operator! Available Operators: + - * /
2.0 \ 5 \ 6.0 = ?
Type an operation: 3 * 4
 3.0 * 4.0 = 12.0
 Type an operation: 6 / 2
 6.0 / 2.0 = 3.0
Would you like to exit?[Y-n]: y
→ calculator git:(master) X
                                                                 Q
                                      dart main.dart
→ calculator git:(master) X dart main.dart
Type an operation: 99 - 1
99.0 - 1.0 = 98.0
Type an operation: -1 - 10
-1.0 - 10.0 = -11.0
Type an operation: -9 + -1
-9.0 + -1.0 = -10.0
Type an operation: -9 + 10
-9.0 + 10.0 = 1.0
Would you like to exit?[Y-n]: n
Type an operation: 25 * 4.5
25.0 * 4.5 = 112.5
Type an operation: 120 / 0.5
120.0 / 0.5 = 240.0
Type an operation: 365 * 999999.5
365.0 * 999999.5 = 364999817.5
```

Kodlar

```
// main.dart
import 'dart:io';
```

```
void main() {
  runApp();
void runApp() {
  for (int i = 0;; i++) {
    Map<String, dynamic> input = getOperation();
    dynamic result =
        calc(input['number1'], input['operator'], input['number2']);
    print(
        "${input['number1']} ${input['operator']} ${input['number2']} =
$result");
    if (askToExit(i)) break;
  }
Map<String, dynamic> getOperation() {
  stdout.write("Type an operation: ");
  List<String>? input = stdin.readLineSync()?.split(' ');
  if (input![1] == '/' && input[2] == '0') {
    print("Numbers can't divide to zero! Try again with another numbers!");
    return getOperation();
  }
  try {
    return {
      'number1': double.parse(input[0]),
      'operator': input[1],
      'number2': double.parse(input[2]),
    };
  } catch (Exception) {
    print(
        "You typed in wrong format! Try again! Format: {number1} {operator}
{number2}");
    return getOperation();
  }
dynamic calc(double number1, String operator, double number2) {
  switch (operator) {
    case '+':
      return number1 + number2;
    case '-':
      return number1 - number2;
    case '*':
      return number1 * number2;
    case '/':
      return number1 / number2;
```

```
default:
    print("You typed unavailable operator! Available Operators: + - * /");
    return '?';
}

bool askToExit(int i) {
    if (i == 0 || (i % 3).floor() != 0) return false;

    stdout.write("Would you like to exit?[Y-n]: ");
    String res = stdin.readLineSync()?.toUpperCase() ?? 'N';

    return res == 'Y';
}
```