

ใบงานที่ 3: การสร้างเครื่องคิดเลขพื้นฐานด้วย React Native

จุดประสงค์การเรียนรู้

1. ได้ฝึกปฏิบัติจริง ในการพัฒนาแอปพลิเคชันด้วย React Native
2. เข้าใจการจัดการอินพุตของผู้ใช้ และการเขียน โปรแกรมแบบขับเคลื่อนด้วยเหตุการณ์ (Event-driven programming)
3. สามารถพัฒนาและจัดการตรรกะคำนวณเลขคณิต รวมถึงจัดการกรณีขอบ (edge cases)
4. ฝึกฝนทักษะการทดสอบและการดีบัก (Debugging) อย่างเป็นระบบ

รายละเอียดการพัฒนาโปรเจกต์

- สร้างแอปพลิเคชัน React Native ที่มีคุณสมบัติดังนี้:
 - รับค่าตัวเลข 2 ค่า ผ่าน TextInput
 - มีปุ่มสำหรับดำเนินการ: บวก, ลบ, คูณ, หาร
 - แสดงผลลัพธ์ผ่าน Alert หรือส่วนแสดงผลอื่นที่เหมาะสม
 - จัดการกรณีอินพุตไม่ถูกต้อง หรือการหารด้วยศูนย์

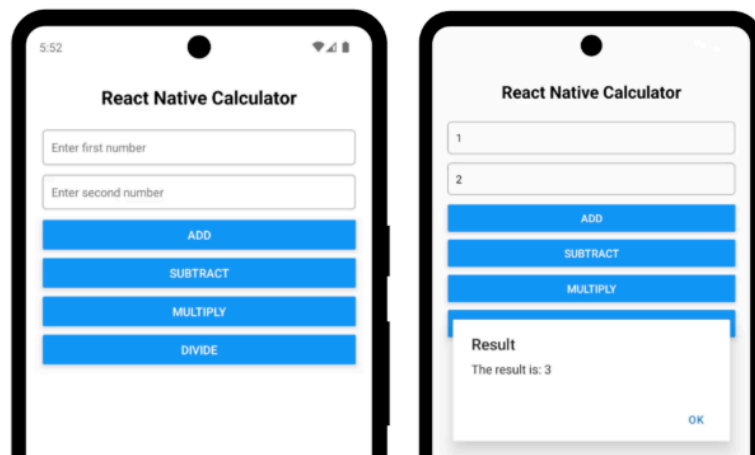
ตัวชี้วัดผลการประเมิน

ส่วนติดต่อผู้ใช้ (User Interface)

- ช่องกรอกข้อมูล:
 - TextInput สองช่องสำหรับกรอกตัวเลข
 - ตรวจสอบความถูกต้องของอินพุตให้เป็นตัวเลขเท่านั้น
- ปุ่มกด (Buttons):
 - มีปุ่ม "Add", "Subtract", "Multiply", "Divide"
 - ปุ่มมีป้ายชื่อชัดเจน และตอบสนองต่อการกด
- การออกแบบ:
 - รูปแบบเรียบง่าย ใช้งานได้จริง
 - ใช้ StyleSheet สำหรับกำหนดรูปแบบที่สอดคล้องกัน

ฟังก์ชันการทำงาน (Functionality)

- การดำเนินการหลัก:
 - คำนวณค่าตาม que ผู้ใช้เลือก
 - จัดการกรณีหารด้วยศูนย์ด้วยการแจ้งเตือน
- การจัดการข้อผิดพลาด:
 - แสดงข้อความเมื่ออินพุตไม่ใช่ตัวเลข



กรณีทดสอบ (Test Cases)

Test Case	Input 1	Input 2	Operation	Expected Result
Addition	10	5	Add	The result is: 15
Subtraction	20	8	Subtract	The result is: 12
Multiplication	4	3	Multiply	The result is: 12
Division	50	5	Divide	The result is: 10
Division by Zero	7	0	Divide	Error: Division by zero is not allowed
Invalid Input	abc	10	Add	Error: Please enter valid numbers
Negative Numbers	-4	-6	Multiply	The result is: 24
Decimal Numbers	5.5	3.2	Add	The result is: 8.7

ส่วนที่ต้องจัดส่ง

1. สร้างไฟล์ PDF ชื่อ worksheet-03-รหัสนักศึกษา.pdf
2. คัดลอกโค้ดทั้งหมด
3. แนบภาพ capture หน้าจอการทำงานของแอปพลิเคชัน (ผลการกดปุ่มแต่ละตัว)

Complete Code and capture all cases

```
import { useState } from 'react';
import { View, Text, TextInput, TouchableOpacity, StyleSheet, Alert } from 'react-native';
```

```
const App = () => {
  const [num_in1, setNum_in1] = useState("");
  const [num_in2, setNum_in2] = useState("");
  const [result, setResult] = useState(null);

  const calculatorAdd = () => {
    const num_cal1 = parseFloat(num_in1);
    const num_cal2 = parseFloat(num_in2);
    if (!isNaN(num_cal1) && !isNaN(num_cal2)) {
      const resultCal = num_cal1 + num_cal2;
      Alert.alert('Result', `The result is: ${resultCal}`);
      setResult(resultCal);
    } else {
      setResult(null);
      Alert.alert('Error', 'Error: Please enter valid numbers');
    }
  };

  const calculatorSubtract = () => {
    const num_cal1 = parseFloat(num_in1);
    const num_cal2 = parseFloat(num_in2);
    if (!isNaN(num_cal1) && !isNaN(num_cal2)) {
      const resultCal = num_cal1 - num_cal2;
      Alert.alert('Result', `The result is: ${resultCal}`);
      setResult(resultCal);
    } else {
      setResult(null);
      Alert.alert('Error', 'Error: Please enter valid numbers');
    }
  };

  const calculatorMultiply = () => {
    const num_cal1 = parseFloat(num_in1);
    const num_cal2 = parseFloat(num_in2);
    if (!isNaN(num_cal1) && !isNaN(num_cal2)) {
      const resultCal = num_cal1 * num_cal2;
      Alert.alert('Result', `The result is: ${resultCal}`);
      setResult(resultCal);
    } else {
      setResult(null);
      Alert.alert('Error', 'Error: Please enter valid numbers');
    }
  };
};
```

```

    }
  };
  const calculatorDivide = () => {
    const num_cal1 = parseFloat(num_in1);
    const num_cal2 = parseFloat(num_in2);
    if (!isNaN(num_cal1) && !isNaN(num_cal2)) {
      const resultCal = num_cal1 / num_cal2;
      if (num_cal2 === 0) {
        Alert.alert('Error', 'Error: Division by zero is not allowed');
        setResult(null);
      } else if (num_cal2 !== 0) {
        Alert.alert('Result', `The result is: ${resultCal}`);
        setResult(resultCal);
      }
    } else {
      setResult(null);
      Alert.alert('Error', 'Error: Please enter valid numbers');
    }
  };

  return (
    <View style={styles.container}>
      <Text style={styles.title}>React Native Calculator</Text>
      <TextInput
        style={styles.input}
        placeholder='Enter first number'
        value={num_in1}
        onChangeText={setNum_in1}
        keyboardType='numeric'
      />
      <TextInput
        style={styles.input}
        placeholder='Enter second number'
        value={num_in2}
        onChangeText={setNum_in2}
        keyboardType='numeric'
      />
      <TouchableOpacity style={styles.button} onPress={calculatorAdd}>
        <Text style={styles.buttonText}>ADD</Text>
      </TouchableOpacity>
      <TouchableOpacity style={styles.button} onPress={calculatorSubtract}>
        <Text style={styles.buttonText}>SUBTRACT</Text>
      </TouchableOpacity>
      <TouchableOpacity style={styles.button} onPress={calculatorMultiply}>
        <Text style={styles.buttonText}>MULTIPLY</Text>
      </TouchableOpacity>
      <TouchableOpacity style={styles.button} onPress={calculatorDivide}>
        <Text style={styles.buttonText}>DIVISION</Text>
    </View>
  );

```

```
    </TouchableOpacity>

    </View>
  );

};

const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    alignItems: 'center',
    padding: 20,
    backgroundColor: '#F9F3EF',
  },
  title: {
    color: '#1B3C53',
    fontSize: 24,
    fontWeight: 'bold',
    marginBottom: 20,
  },
  input: {
    width: '80%',
    height: 40,
    borderColor: 'gray',
    backgroundColor: 'white',
    borderWidth: 1,
    borderRadius: 5,
    paddingHorizontal: 10,
    marginBottom: 20,
  },
  button: {
    backgroundColor: '#456882',
    padding: 10,
    marginBottom: 10,
    borderRadius: 5,
    width: '80%',
    alignItems: 'center',
  },
  buttonText: {
    color: 'white',
    fontSize: 18,
  },
  result: {
    marginTop: 20,
    fontSize: 18,
    fontWeight: 'bold',
  },
});
```

66106030 บัญชา วาสนสิทธิ์

```
});
```

```
export default App;
```

66106030 บัญชา วัฒนสิทธิ์

Capture Result

