

# การรับและแสดงผลข้อมูล

## Data Input and Output

ผู้ช่วยศาสตราจารย์สมเกียรติ ช่อเหมือน (tko@webmail.npru.ac.th)

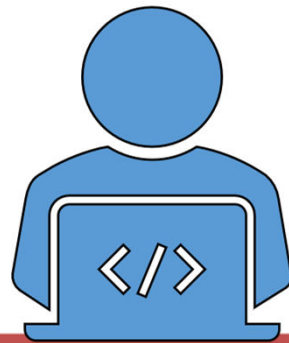
สาขาวิชาวิศวกรรมซอฟต์แวร์ คณะวิทยาศาสตร์และเทคโนโลยี



# บทนำ



- วิธีการรับข้อมูลจากผู้ใช้
  - การใช้งานฟังก์ชันสำหรับรับข้อมูล
  - การจัดการข้อมูล: การตรวจสอบความถูกต้อง การแปลงชนิดข้อมูล
- . วิธีการแสดงผลข้อมูลให้กับผู้ใช้
  - การใช้งานฟังก์ชันสำหรับแสดงผลข้อมูล
  - การจัดรูปแบบข้อมูลเมื่อแสดงผล: การจัดเรียงข้อความ การปรับแต่งรูปแบบตัวเลข



## การใช้งานฟังก์ชันสำหรับรับข้อมูล:

- ใน JavaScript สามารถใช้รับข้อมูลจากผู้ใช้
- การใช้งานฟังก์ชัน `prompt()` ซึ่งจะแสดงหน้าต่างป๊อปอัปให้ผู้ใช้

```
let userInput = prompt("Please enter your name:");  
// แสดงหน้าต่างป๊อปอัปและเก็บข้อมูลที่ผู้ใช้ป้อน  
console.log("Hello, " + userInput + "!");  
// แสดงข้อความทักทายพร้อมชื่อที่ผู้ใช้ป้อน
```

# การจัดการข้อมูล

- การตรวจสอบความถูกต้อง (Validation)

```
let ageInput = prompt("Please enter your age:");  
if (isNaN(ageInput)) {  
  // ตรวจสอบว่าข้อมูลที่ป้อนเป็นตัวเลขหรือไม่  
  console.log("Invalid input. Please enter a number.");  
} else {  
  console.log("Your age is: " + ageInput);  
}
```

- การแปลงชนิดข้อมูล (Type Conversion)

```
let stringNumber = "123";  
let convertedNumber = Number(stringNumber);  
// แปลงข้อความเป็นตัวเลข  
console.log(convertedNumber + 10); // Output: 133
```

---

# วิธีการแสดงผลข้อมูลให้กับผู้ใช้

- การใช้งานฟังก์ชันสำหรับแสดงผลลัพธ์
- การจัดรูปแบบข้อมูลเพื่อนำเสนอ



# ฟังก์ชันสำหรับแสดงผล

- การแสดงผลข้อมูลใน JavaScript

- การใช้ **console.log()** แสดงผลข้อมูลใน console ของเบราว์เซอร์.

```
let name = "John";
```

```
console.log("Hello, " + name + "!"); // Output: Hello, John!
```

- การใช้ **alert()** ใช้สำหรับแสดงผลข้อมูลในหน้าต่าง popup.

```
alert("This is an alert message!");
```

- การใช้ **document.write()** ใช้สำหรับแสดงผลข้อมูลบนหน้าเว็บ.

- `document.write("This text is displayed on the webpage.");`



# การจัดรูปแบบข้อมูลเมื่อแสดงผล



- **การจัดเรียงข้อความ (String Formatting)**
- การใช้ `template literals` ในการจัดรูปแบบข้อความ
  - `let firstName = "John";`
  - `let lastName = "Doe";`
  - `console.log(`Hello, my name is ${firstName} ${lastName}.`);` // Output: Hello, my name is John Doe
- **การปรับแต่งรูปแบบตัวเลข (Number Formatting)**
- การใช้ `toLocaleString()` หรือ `toFixed()` ในการจัดรูปแบบตัวเลข
  - `let num = 123456.789;`
  - `console.log(num.toLocaleString());` // Output: 123,456.789 (ในรูปแบบของประเทศท้องถิ่น)
  - `console.log(num.toFixed(2));` // Output: 123456.79 (เป็นตัวเลขทศนิยม 2 ตำแหน่ง)

# สรุปท้ายบท

- การรับข้อมูลจากผู้ใช้สามารถทำได้ผ่านหลายวิธี เช่น `prompt()` `alert()`
- การตรวจสอบความถูกต้อง เช่น เงื่อนไขหรือการแปลงชนิดข้อมูล
- การจัดรูปแบบข้อมูล เช่น การจัดเรียงข้อความหรือฟังก์ชัน `toLocaleString()`
- การสร้างปฏิสัมพันธ์กับผู้ใช้และนำเสนอข้อมูลที่สื่อความหมาย





# ถาม-ตอบ

