

Alphonz George

T11 03

ASSIGNMENT 5

Aim: To study conditional Statements, Loops, Functions, Inheritance, Iterators and Generators.

Code:

// 1. Conditional Statements (if-else, switch) function

```
checkNumber(num) {  
  if (num > 0) {  
    console.log(`${num} is positive.`);  
  } else if (num < 0) {  
    console.log(`${num} is negative.`);  
  } else {  
    console.log(`${num} is zero.`);  
  }  
}
```

// Using switch statement function

```
getDayName(dayNumber) {  
  switch (dayNumber) {  
    case 1:  
      return 'Monday';  
    case 2:  
      return 'Tuesday';  
    case 3:
```

```
        return 'Wednesday';
case 4:
    return 'Thursday';
case 5:
    return 'Friday';
case 6:
    return 'Saturday';
case 7:
    return 'Sunday';
default:
    return 'Invalid day number';
    }
}
```

// 2. Loops (for, while) function

```
printNumbersWithForLoop(n) {
  console.log('Using for loop:');  for (let i
= 1; i <= n; i++) {    console.log(i);
  }
}
```

```
function printNumbersWithWhileLoop(n) {
  console.log('Using while loop:');
```

```
    let i = 1;    while
(i <= n) {
console.log(i);
i++;
    }
}
```

// 3. Functions (regular and arrow functions)

```
function add(a, b) {    return a + b;
}
```

```
const multiply = (a, b) => a * b;
```

// 4. Inheritance (using ES6 classes)

```
class Animal {    constructor(name)
{        this.name = name;
    }
```

```
    speak() {
        console.log(` ${this.name} makes a sound.`);
    }
}
```

```

class Dog extends Animal {
  constructor(name, breed) {
    super(name); // calling the parent class constructor
    this.breed = breed;
  }

  speak() {
    console.log(`${this.name} barks. It's a ${this.breed}.`);
  }
}

```

```

// 5. Iterators (custom iterator)
const iterableObject = {
  values: [1, 2, 3, 4, 5],
  [Symbol.iterator]() {
    let
    index = 0;
    return {
      next: () => {
        if (index < this.values.length) {
          return { value: this.values[index++], done: false };
        } else {
          return { done: true };
        }
      }
    };
  }
};

```

```
    }  
};
```

```
// 6. Generators (using `function*`) function*
```

```
numberGenerator() {
```

```
    let i = 1;
```

```
    while (i <= 5) {
```

```
        yield i++;
```

```
    }
```

```
}
```

```
// Running the examples console.log("===
```

```
Conditional Statements ===");
```

```
checkNumber(5);    // 5 is positive.
```

```
checkNumber(-3);   // -3 is negative.
```

```
checkNumber(0);    // 0 is zero.
```

```
console.log(getDayName(3)); // Wednesday
```

```
console.log("\n=== Loops ===");
```

```
printNumbersWithForLoop(5); // 1 2 3 4 5
```

```
printNumbersWithWhileLoop(5); // 1 2 3 4 5
```

```
console.log("\n=== Functions ==="); console.log(`5 + 3 =
```

```
${add(5, 3)}`);    // 5 + 3 = 8 console.log(`5 * 3 =
```

```
${multiply(5, 3)}`); // 5 * 3 = 15
```

```
console.log("\n=== Inheritance ==="); const  
animal = new Animal("Generic Animal");  
animal.speak(); // Generic Animal makes a sound.
```

```
const dog = new Dog("Max", "Golden Retriever");  
dog.speak(); // Max barks. It's a Golden Retriever.
```

```
console.log("\n=== Iterators ==="); for  
(let value of iterableObject) {  
  console.log(value); // 1 2 3 4 5  
}
```

```
console.log("\n=== Generators ===");  
const gen = numberGenerator(); for  
(let value of gen) {  
  console.log(value); // 1 2 3 4 5  
}
```

Output:

=== Conditional Statements ===	book.js:106
5 is positive.	book.js:4
-3 is negative.	book.js:6
0 is zero.	book.js:8
Wednesday	book.js:110
	book.js:112
=== Loops ===	
Using for loop:	book.js:36
1	book.js:38
2	book.js:38
3	book.js:38
4	book.js:38
5	book.js:38
Using while loop:	book.js:43
1	book.js:46
2	book.js:46
3	book.js:46
4	book.js:46
5	book.js:46

=== Functions ===
5 + 3 = 8
5 * 3 = 15
=== Inheritance ===
Generic Animal makes a sound.
Max barks. It's a Golden Retriever.
=== Iterators ===
1
2
3
4
5
=== Generators ===
1
2
3
4
5