# DAY-19 EVENING ASSESSMENT

# VARIABLES

1: Declare a variable age of type number and assign it a value. Print it.

let age = 25;  
console.log(age);  
  
2: Create a variable username of type string and log "Hello, <username>"

let username = "Sri Vidya";  
console.log("Hello, " + username);  
  
3: Declare a boolean variable isActive and assign it true. Print its type.

let isActive = true;  
console.log(typeof isActive);  
  
4: Create two number variables x and y, assign values, and print their sum.

let x = 10, y = 20;  
console.log(x + y);  
  
5: Declare a variable colors as an array of strings with three colors. Print the second one.

let colors = ["Red", "Blue", "Green"];  
console.log(colors[1]);  
  
6: Create a constant PI with value 3.14 and try reassigning it (observe the error).

const PI = 3.14;  
console.log(PI);  
// PI = 3.14159; // Uncomment to see error

Functions   
  
7: Write a function greet that takes a name (string) and returns "Hello, <name>"

function greet(name) {  
 return "Hello, " + name;  
}  
console.log(greet("Sri Vidya"));  
  
8: Write a function addNumbers that takes two numbers and returns their sum.

function addNumbers(a, b) {  
 return a + b;  
}  
console.log(addNumbers(5, 7));  
  
9: Create a function isEven that takes a number and returns true if it’s even, else false.

function isEven(num) {  
 return num % 2 === 0;  
}  
console.log(isEven(4));  
console.log(isEven(7));  
  
10: Write a function multiply with default parameter b = 5 that multiplies a \* b.

function multiply(a, b = 5) {  
 return a \* b;  
}  
console.log(multiply(4));  
console.log(multiply(4, 3));  
  
11: Create an arrow function square that takes a number and returns its square.

const square = num => num \* num;  
console.log(square(6));  
  
12: Write a function printDetails that accepts a name (string) and age (number) and prints:

function printDetails(name, age) {  
 console.log(`Name: ${name}, Age: ${age}`);  
}  
printDetails("Sri Vidya", 23);  
  
Classes   
  
13: Create a class Person with name and age properties, and a method introduce()

class Person {  
 introduce() {  
   console.log(`Hi, I'm ${this.name} and I'm ${this.age} years old.`);  
 }  
}  
  
14: Add a constructor to Person that initializes name and age.

class PersonWithConstructor {  
 constructor(name, age) {  
   this.name = name;  
   this.age = age;  
 }  
 introduce() {  
   console.log(`Hi, I'm ${this.name} and I'm ${this.age} years old.`);  
 }  
}  
const person1 = new PersonWithConstructor("Sri Vidya", 23);  
person1.introduce();  
  
15: Create a class Car with properties brand and year, and a method displayInfo()

class Car {  
 constructor(brand, year) {  
   this.brand = brand;  
   this.year = year;  
 }  
 displayInfo() {  
   console.log(`Car: ${this.brand}, Year: ${this.year}`);  
 }  
}  
const car1 = new Car("Toyota", 2022);  
car1.displayInfo();

16: Create a class Rectangle with properties width and height and a method getArea()

class Rectangle {  
 constructor(width, height) {  
   this.width = width;  
   this.height = height;  
 }  
 getArea() {  
   return this.width \* this.height;  
 }  
}  
const rect1 = new Rectangle(5, 10);  
console.log(rect1.getArea());  
  
17: Create a class Student that has name and grade, and a method displayGrade()

class Student {  
 constructor(name, grade) {  
   this.name = name;  
   this.grade = grade;  
 }  
 displayGrade() {  
   console.log(`Student ${this.name} has grade ${this.grade}`);  
 }  
}  
const student1 = new Student("Anjali", "A");  
student1.displayGrade();  
  
18: Create a class BankAccount with accountNumber and balance, and a method deposit()

class BankAccount {  
 constructor(accountNumber, balance) {  
   this.accountNumber = accountNumber;  
   this.balance = balance;  
 }  
 deposit(amount) {  
   this.balance += amount;  
   console.log(`New balance: ${this.balance}`);  
 }  
}  
const account1 = new BankAccount("123456789", 5000);  
account1.deposit(1500);