Given an object:

```
var person = {
        "firstName": "Harry",
        "lastName": "Potter",
        "age": 30,
        "gender": "male",
        "skill": "ReactJS",
        "expertise": "Beginner",
       };
```

1. Print the firstname.
2. Print the lastname.
3. Print the fullname("Harry Potter").
4. If the age is less than 18, then print "false". If the age is more than 18, then print "true".

Assignment 2:

```
const person = {
        name: "Alice",
        age: 30,
        city: "New York",
        hobbies: ["reading", "hiking", "coding"],
        isStudent: false
       };
```

1. What is Alice's age?
2. How would you change Alice's city to "Los Angeles"?
3. How would you add a new property called "occupation" with the value "Software Engineer"?
4. How would you loop through all of Alice's hobbies?
5. How would you check if the object has a property called "occupation"?
6. What is the data type of the `isStudent` property?

Assignment 3:

```
const person = {
        name: "Alice",
        age: 30,
        city: "New York",
        hobbies: ["reading", "hiking", "coding"],
        isStudent: false
       };
```

Basic Level

1. **Accessing Properties:**
   - Write a JavaScript statement to print Alice's age from the `person` object.
   - How would you change Alice's `city` to `"San Francisco"`?
2. **Adding & Removing Properties:**
   - Add a new property `occupation` with the value `"Software Engineer"`.
   - Remove the `isStudent` property from the `person` object.
3. **Array Operations:**
   - Add `"swimming"` to Alice's hobbies list.
   - Remove the last hobby from the array.
4. **Looping Through Object:**
   - Write a loop to print all properties and their values in the `person` object.

Intermediate Level

5. **Object Methods:**
   - Add a method `greet()` to `person` that returns `"Hello, my name is Alice!"`.
   - Modify `greet()` to dynamically return `"Hello, my name is [name] and I am [age] years old!"`.
6. **Object Destructuring:**
   - Use object destructuring to extract `name` and `age` from `person`.
7. **Using `this` in an Object Method:**
   - Add a method `displayHobbies()` that logs all hobbies in a formatted string.

Advanced Level

8. **Object Copying & Spread Operator:**
   - Create a new object `newPerson` by copying `person` but changing the `age` to `35`.
9. **Checking Property Existence:**
   - Write a function that checks if a given property exists in the `person` object.
10. **Merging Objects:**
    - Merge `person` with another object `{ country: "USA", language: "English" }`.

Assignment 4:

Here's a more **complex object** with nested structures, arrays, and methods:

```
const person = {
        name: "Alice",
        age: 30,
        address: {
          street: "123 Main St",
          city: "New York",
          country: "USA"
        },
        hobbies: ["reading", "hiking", "coding"],
        isStudent: false,
        contacts: [
          { type: "email", value: "alice@example.com" },
          { type: "phone", value: "+1234567890" }
        ],
        skills: {
          programming: ["JavaScript", "Python", "Java"],
          softSkills: ["communication", "teamwork"]
        },
        greet: function () {
          return `Hello, my name is ${this.name} and I live in ${this.address.city}.`;
        },
        addHobby: function (newHobby) {
          this.hobbies.push(newHobby);
        }
      };
```

Assignment Questions:

Basic Level

1. Print Alice's city from the `address` object.
2. Add a new contact `{ type: "LinkedIn", value: "linkedin.com/alice" }` to the `contacts` array.
3. Change Alice's country to `"Canada"`.
4. Print the first programming language Alice knows.
5. Call the `greet()` method and display its result.

Intermediate Level

6. Add a method `displayContacts()` that logs all contact types and values.
7. Write a function that adds a new skill under `skills.programming`.
8. Write a function to remove a hobby from the `hobbies` array.
9. Loop through `skills.programming` and print each programming language.
10. Use object destructuring to extract `name` and `age`.

Advanced Level

11. Create a deep copy of the `person` object without modifying the original.
12. Write a function to find a contact by type (e.g., `"email"`) and return its value.
13. Merge `person` with another object `{ job: "Software Developer", experience: 5 }`.
14. Check if `skills.softSkills` contains `"leadership"` and add it if not.
15. Convert the `person` object to a JSON string and parse it back into an object.