**Условия и ветвления: if, else, switch**

**3.1. Зачем нужны условия?**

Иногда программа должна **выполнять разные действия в зависимости от ситуации**:

* Если пользователь ввёл пароль — пускаем на сайт.
* Если число больше 0 — пишем "положительное".
* Если нажали на кнопку "Удалить" — спрашиваем подтверждение.

👉 Для этого в JavaScript есть конструкции **ветвления**: if, else, else if, switch.

**3.2. Конструкция if**

js

if (условие) {

// код, если условие ИСТИННО (true)

}

Пример:

js

КопироватьРедактировать

let age = 18;

if (age >= 18) {

console.log("Ты взрослый");

}

**3.3. if...else**

js

if (условие) {

// если true

} else {

// если false

}

Пример:

js

let loggedIn = false;

if (loggedIn) {

console.log("Добро пожаловать!");

} else {

console.log("Войдите в систему");

}

**3.4. if...else if...else**

js

if (a > 10) {

// условие 1

} else if (a > 5) {

// условие 2

} else {

// всё остальное

}

**3.5. Сравнения и логика**

| **Оператор** | **Что делает** |
| --- | --- |
| == | сравнение по значению (не строго) |
| === | строгое сравнение (и тип, и значение) |
| != | не равно |
| !== | строго не равно |
| > < >= <= | больше, меньше и т.п. |

**3.6. Логические операторы**

| **Оператор** | **Назначение** | **Пример** |
| --- | --- | --- |
| && | И (оба условия true) | age > 18 && male |
| ` |  | ` |
| ! | НЕ (инверсия) | !isAdmin |

**3.7. Тернарный оператор**

js

условие ? значение\_если\_true : значение\_если\_false;

Пример:

js

let access = age >= 18 ? "Доступ разрешён" : "Отказано";

**3.8. Конструкция switch**

Когда нужно проверить одно значение на множество вариантов:

js

let role = "admin";

switch (role) {

case "admin":

console.log("Панель администратора");

break;

case "user":

console.log("Панель пользователя");

break;

default:

console.log("Неизвестная роль");

}

**3.9. Вложенные условия**

js

if (isOnline) {

if (isAdmin) {

console.log("Панель админа");

} else {

console.log("Панель юзера");

}

}

**3.10. Лучшие практики**

✅ Пиши условия понятно и читаемо  
✅ Используй === вместо ==  
✅ Не пиши слишком много вложенных if  
✅ Предпочитай тернарный оператор для коротких выражений  
✅ switch — для чётких альтернатив

**3.11. Частые ошибки**

❌ Забыли break в switch — проваливание в другие case  
❌ Сравниваете 10 === "10" — false, потому что типы разные  
❌ Путаете = и ==  
❌ Пропущены фигурные скобки {}  
❌ Сложные вложенные if, где лучше switch или логические операторы

**3.12. Глоссарий**

| **Термин** | **Значение** |
| --- | --- |
| Ветвление | Путь выполнения, зависящий от условий |
| if | Проверка условия |
| else | Альтернатива при ложном условии |
| switch | Альтернатива множеству else if |
| break | Прерывание switch |
| Тернарный | Короткий if-else в одну строку |

**3.13. Вопросы для самопроверки**

1. В чём разница между == и ===?
2. Когда использовать switch, а когда if?
3. Что делает оператор &&?
4. Почему важно ставить break в switch?
5. Что вернёт тернарный оператор: 5 > 3 ? "Да" : "Нет"?