**Лабораторная работа №2**

**Тема работы:** Разработка программ с применением явного и неявного приведения типов данных.

**Цель работы:** изучить способы работы и способы использования явного и неявного приведения типов.

**Порядок выполнения работы**

1. Изучить краткие теоретические сведения
2. Выполнить практические задания
3. Ответить на контрольные вопросы
4. Сдать преподавателю отчет в форме:
   1. Листинг программы;
   2. Скриншот результата.

**Краткие теоретические сведения**

**Конструкция if-else**

Для того, чтобы напрограммировать что-нибудь **полезное**, одних переменных далеко не достаточно. Нам нужен механизм, который позволит выполнять *определенный код в зависимости от каких-либо условий*.

То есть нам нужно иметь возможность спросить у JavaScript **'Если'**.

Например **так:** *если эта переменная меньше нуля, то вывести 'отрицательно', иначе (то есть если она больше нуля) вывести 'положительно'.*

В JavaScript для таких вопросов предназначена **конструкция if**, которая позволяет выполнять **определенный код при выполнении какого-либо условия**:

if (логическое выражение) {

Этот код выполниться,

если логическое выражение верно (то есть равно true)} else {

Этот код выполнится,

если логическое выражение неверно (то есть равно false)}

Обратите внимание на то, что **блок else не обязателен**.

Логическое **выражение** представляет собой тот самый вопрос, который мы хотим задать JavaScript. Например, чтобы спросить *'переменная* ***a*** *больше нуля'* мы напишем так: **a > 0**.

Примеры работы:

var a **=** 3;

/\*

Если переменная a больше нуля, то выведи 'верно',

иначе (если меньше или равна нулю) выведи 'неверно'

\*/if (a **>** 0) {alert('Верно!');} else {alert('Неверно!');} //выведет 'Верно!'

var a **=** **-**3;

/\*

Если переменная a больше или равна нулю, то выведи 'верно',

иначе (если меньше нуля) выведи 'неверно'

\*/if (a **>=** 0) {alert('Верно!');} else {alert('Неверно!');} //выведет 'Неверно!'

## Сокращенный синтаксис

В случае, если в фигурных скобках **if** или **else** будет **только одно выражение**, можно эти фигурные скобки не писать:

//Полный вариант:if (a **==** 0) {alert('Верно!');} else {alert('Неверно!');}

//Уберем скобки после if:if (a **==** 0) alert('Верно!'); else {alert('Неверно!');}

//Уберем скобки после else:if (a **==** 0) {alert('Верно!');} else alert('Неверно!');

/\*

Уберем скобки и после if, и после else

(обратите внимание на точку с запятой - она осталась):

\*/if (a **==** 0) alert('Верно!'); else alert('Неверно!');

## Равенство по значению и типу

Для того, чтобы сравнить на **равенство** следует использовать оператор двойное равно **==**, а не одиночное **=**, как можно было подумать.

Почему **так?** Потому что **одиночное равно зарезервировано за присваиванием**. Смотрите пример:

var a **=** 0;

/\*

Если переменная a равна нулю, то выведи 'верно',

иначе (если не равна нулю) выведи 'неверно'

\*/if (a **==** 0) alert('Верно!'); else alert('Неверно!'); //выведет 'Верно!'

А следующий пример работать будет **не так**, как мы думаем:

var a **=** 0;

/\*

Мы думаем оно работает так:

если переменная a равна нулю, то выведи 'верно',

иначе (если не равна нулю) выведи 'неверно'.

На самом деле оно работает так:

переменной a присвоить 1,

если удалось присвоить - то выведи 'верно',

иначе (если не удалось присвоить) выведи 'неверно'.

\*/

if (a **=** 1) alert('Верно!'); else alert('Неверно!'); //всегда будет выводить 'Верно!'

Кроме оператора **==** существует еще и оператор **===**. Их различие в том, что **===** сравнивает не только по значению, но и **по типу**, а **==** сравнивает только по значению.

Чтобы полностью разобраться в этом **внимательно изучите примеры**:

var a **=** '0'; //переменная a представляет собой строку, а не число 0if (a **==** 0) alert('Верно!'); else alert('Неверно!');

/\*

Выведет 'Верно!', так как проверяется только значение, но не тип.

Поэтому '0' равен 0.

\*/

var a **=** '0'; //переменная a представляет собой строку, а не число 0if (a **===** 0) alert('Верно!'); else alert('Неверно!');

/\*

Выведет 'Неверно!', так как строка '0'

не равна числу 0 при сравнении по типу.

\*/

## Не равно

Для того, чтобы спросить **'не равно'**, существует операторы **!=** и **!==**. **Первый** игнорирует различие в типах, а **второй** - нет.

var a **=** 0;

/\*

Если переменная a НЕ равна нулю, то выведи 'верно',

иначе (если равна нулю) выведи 'неверно'

\*/if (a **!=** 0) alert('Верно!'); else alert('Неверно!'); //выведет 'Неверно!', так как a равна 0

a **=** 1;

/\*

Если переменная a НЕ равна нулю, то выведи 'верно',

иначе (если равна нулю) выведи 'неверно'

\*/if (a **!=** 0) alert('Верно!'); else alert('Неверно!'); //выведет 'Верно!', так как a равна 1

var a **=** '0';

/\*

Если переменная a НЕ равна нулю, то выведи 'верно',

иначе (если равна нулю) выведи 'неверно'

\*/if (a **!=** 0) alert('Верно!'); else alert('Неверно!');

/\*

Выведет 'Неверно!', так как a равно '0',

а различие в типах игнорируется.

\*/

var a **=** '0';

/\*

Если переменная a НЕ равна нулю, то выведи 'верно',

иначе (если равна нулю) выведи 'неверно'

\*/if (a **!==** 0) alert('Верно!'); else alert('Неверно!');

/\*

Выведет 'Верно!', так как a равно '0',

а это не равно 0 при сравнении по типу.

\*/

## Все операции сравнения

Возможные операции сравнения, которые можно использовать внутри **if**:

| a == b | **a** равно **b** |
| --- | --- |
|  |  |
| a === b | **a** равно **b** и они одинаковы по типу |
|  |  |
| a != b | **a** не равно **b** |
|  |  |
| a !== b | **a** не равно **b** или **a** равно **b**, но ни разные по типу |
|  |  |
| a < b | **a** меньше **b** |
|  |  |
| a > b | **a** больше **b** |
|  |  |
| a <= b | **a** меньше или равно **b** |
|  |  |
| a >= b | **a** больше или равно **b** |
|  |  |

## Несколько условий сразу

Иногда может быть нужно составить какое-то **сложное условие**, например, пользователь вводит месяц своего рождения и вам нужно проверить, что введенное число *больше или равно 1 и меньше либо равно 12* (так как в году 12 месяцев).

Для этого существуют операторы **&&** (логическое И) и **||** (логическое ИЛИ).

var a **=** 3;var b **=** **-**3;//Если a больше нуля и b одновременно меньше нуля то...if (a **>** 0 **&&** b **<** 0) alert('Верно!'); else alert('Неверно!'); //выведет 'Верно!'

var a **=** 3;//Если a больше или равно 1 и меньше или равно 12 то...if (a **>=** 1 **&&** a **<=** 12) alert('Верно!'); else alert('Неверно!'); //выведет 'Верно!'

var a **=** **-**3;var b **=** **-**3;/\*

Если a больше нуля ИЛИ b меньше нуля - хотя бы один из них, то...

выведет 'Верно!', так как хотя a и не больше нуля,

но одно из условий - b < 0 - выполнится!

\*/if (a **>** 0 **||** b **<** 0) alert('Верно!'); else alert('Неверно!');

## Работа с логическими переменными

Многие функции JavaScript в результате своей работы возвращают либо **true** (истина), либо **false** (ложь). Эти значения довольно удобны при своей работе, но новичкам бывает довольно сложно понять их.

Представим себе, что переменная **a** равна **true**. В таком случае конструкцию **if** можно записать так:

var a **=** true;

//Если a равно true, то...if (a **==** true) alert('Верно!'); else alert('Неверно!');

/\*

Выведет 'Верно!', так как a равно true.

\*/

Так как такие сравнения довольно распространены в JavaScript, то существует специальный прием, облегчающий работу (но не понимание, к сожалению).

Прием такой: конструкцию **a == true** можно заменить на более простую: вместо **if (a == true)** написать **if (a)** и она будет работать аналогично.

Следует *пользоваться второй конструкцией, так как она проще.*

/\*

Заменим a == true на более простую:

вместо if (a == true) напишем if (a):

\*/

var a **=** true;//Если a равно true, то...if (a) alert('Верно!'); else alert('Неверно!'); //выведет 'Верно!', так как a равно true

var a **=** true;//Если a НЕ true (то есть false!), то...if (**!**a) alert('Верно!'); else alert('Неверно!'); //выведет 'Неверно!', так как a равно true

Также обратите внимание на следующие примеры:

//Данное выражение всегда будет выводить 'Верно'if (true) alert('Верно!'); else alert('Неверно!');

//Данное выражение всегда будет выводить 'Неверно'if (false) alert('Верно!'); else alert('Неверно!');

//Данное выражение всегда будет выводить 'Неверно'if (**!**true) alert('Верно!'); else alert('Неверно!');

//Данное выражение всегда будет выводить 'Верно'if (**!**false) alert('Верно!'); else alert('Неверно!');

## Вложенные if

Предположим, нам необходимо спросить у JavaScript такую вещь: *если переменная* ***a*** *не определена, то вывести 'Введите a', если определена, то проверить - больше нуля* ***a*** *или нет. Если больше нуля - то вывести 'Больше нуля!', если меньше - вывести 'Меньше нуля'.*

Одной конструкцией **if** здесь не обойтись, нужно использовать две таким образом, чтобы одна была внутри другой:

if (a **===** undefined) { //если переменная a не определена

alert('Введите a!');} else { //если переменная a НЕ пуста

if (a **>** 0) { //спрашиваем, больше ли нуля переменная a

alert('Больше нуля!');

} else {

alert('Меньше нуля!');

}}

## Конструкция else if

Недостатком конструкции предыдущего примера является большое количество фигурных скобок.

Чтобы избавиться от них, можно пользоваться специальной конструкцией **else if**, которая представляет собой одновременно и **else**, и начало вложенного **if**:

//Решение предыдущей задачи через конструкцию else if:if (a **===** undefined) {

alert('Введите a!');} else if (a **>** 0) {

alert('Больше нуля!');

} else {

alert('Меньше нуля!');}

Можно использовать **несколько else if**, но злоупотреблять этим не стоит (лучше будет воспользоваться конструкцией **switch-case**, о которой ниже).

## Несколько if

Пусть у нас есть такая задача: *сайт поддерживает 3 языка - русский, английский, немецкий. Переменная* ***lang*** *может принимать 3 значения - 'ru', 'en' и 'de'. В зависимости от значения переменной* ***lang*** *следует вывести фразу на одном из языков.*

Решение: можно было бы воспользоваться **вложенными ифами** или **else if**. Выглядело бы это примерно так:

//Решение задачи через else if – не самое удачное:if (lang **==** 'ru') { //фраза на русском

alert('Русский текст');} else if (lang **==** 'en') { //фраза на английском

alert('Английский текст');} else if (lang **==** 'de') { //фраза на немецком

alert('Немецкий текст');}

Такое решение не слишком красивое - блок **else** здесь не нужен! Проще всего будет написать не один длинный **if** с несколькими **else**, а *несколько if вообще без else*:

//Решение задачи через несколько if – оно намного лучше:if (lang **==** 'ru') { //фраза на русском

alert('Русский текст');} if (lang **==** 'en') { //фраза на английском

alert('Английский текст');}if (lang **==** 'de') { //фраза на немецком

alert('Немецкий текст');}

/\*

В данном коде сработает только один из ифов,

так как переменная lang может иметь только одно из значений

\*/

Однако это решение тоже не слишком удобно. Представьте, что у вас будет не три языка, а 10 - вам придется написать 10 конструкций if.

Для таких случаев существует конструкция **switch-case**.

## Конструкция switch-case

Данная конструкция представляет собой альтернативу **if-else**, ее рекомендуется использовать в случае множественного выбора (например, 10 различных языков, как в нашей задаче).

Изучите ее синтаксис:

switch (переменная) {

case 'значение1':

здесь код, который выполнится в случае, если переменная имеет значение1;

break;

case 'значение2':

здесь код, который выполнится в случае, если переменная имеет значение2;

break;

case 'значение3':

здесь код, который выполнится в случае, если переменная имеет значение3;

break;

default:

этот код выполнится в случае, если переменная не совпала ни с одним значением;

break;}

Решим нашу задачу с тремя языками с помощью данной конструкции:

switch (lang) {

case 'ru':

alert('Русский текст');

break;

case 'en':

alert('Английский текст');

break;

case 'de':

alert('Немецкий текст');

break;

default:

alert('Данный язык не поддерживается');

break;}

**Порядок выполнения работы**

**Примеры решения задач**

### Задача №1

**Задача.** Если переменная **a** равна 10, то выведите 'Верно', иначе выведите 'Неверно'.

**Решение:**

var num **=** 10;

if (num **==** 10) {

alert('Верно');

} else {

alert('Неверно');

}

### Задача №2

**Задача.** В переменной **min** лежит число от 0 до 59. Определите в какую **четверть часа** попадает это число (в первую, вторую, третью или четвертую).

**Решение:**

var min **=** 10;

if (min **>=** 0 **&&** min **<=** 14) {

alert('В первую четверть.');

}

if (min **>=** 15 **&&** min **<=** 30) {

alert('Во вторую четверть.');

}

if (min **>=** 31 **&&** min **<=** 45) {

alert('В третью четверть.');

}

if (min **>=** 46 **&&** min **<=** 59) {

alert('В четвертую четверть.');

}

### Задача №3

Переменная **lang** может принимать 2 значения: **'ru' 'en'**. Если она имеет значение 'ru', то в переменную arr запишем массив дней недели на русском языке, а если имеет значение 'en' – то на английском. Решите задачу через **2 if**, через **switch-case** и через **многомерный массив** без ифов и switch.

**Решение:**

Решение через 2 if:

var lang **=** 'ru';

if (lang **==** 'ru') {

var arr **=** ['пн', 'вт', 'ср', 'чт', 'пт', 'сб', 'вс'];

}

if (lang **==** 'en') {

arr **=** ['mn', 'ts', 'wd', 'th', 'fr', 'st', 'sn'];

}

alert(arr);

Решение через switch-case:

var lang **=** 'ru';

switch (lang) {

case 'ru':

var arr **=** ['пн', 'вт', 'ср', 'чт', 'пт', 'сб', 'вс'];

break;

case 'en':

arr **=** ['mn', 'ts', 'wd', 'th', 'fr', 'st', 'sn'];

break;

}

alert(arr);

Решение через многомерный массив:

var lang **=** 'ru';

var arr **=** {

'ru':['пн', 'вт', 'ср','чт', 'пт', 'сб', 'вс'],

'en':['mn', 'ts', 'wd', 'th', 'fr', 'st', 'sn'],

};

alert(arr[lang]);

**Практическая часть**

### Работа с if-else

1. Если переменная **a** равна нулю, то выведите 'Верно', иначе выведите 'Неверно'. Проверьте работу скрипта при **a**, равном 1, 0, -3.
2. Если переменная **a** больше нуля, то выведите 'Верно', иначе выведите 'Неверно'. Проверьте работу скрипта при **a**, равном 1, 0, -3.
3. Если переменная **a** меньше нуля, то выведите 'Верно', иначе выведите 'Неверно'. Проверьте работу скрипта при **a**, равном 1, 0, -3.
4. Если переменная **a** больше или равна нулю, то выведите 'Верно', иначе выведите 'Неверно'. Проверьте работу скрипта при **a**, равном 1, 0, -3.
5. Если переменная **a** меньше или равна нулю, то выведите 'Верно', иначе выведите 'Неверно'. Проверьте работу скрипта при **a**, равном 1, 0, -3.
6. Если переменная **a** не равна нулю, то выведите 'Верно', иначе выведите 'Неверно'. Проверьте работу скрипта при **a**, равном 1, 0, -3.
7. Если переменная **a** равна 'test', то выведите 'Верно', иначе выведите 'Неверно'. Проверьте работу скрипта при **a**, равном 'test', 'тест', 3.
8. Если переменная **a** равна '1' и по значению и по типу, то выведите 'Верно', иначе выведите 'Неверно'. Проверьте работу скрипта при **a**, равном '1', 1, 3.

### Работа с логическими переменными

1. Если переменная **test** равна true, то выведите 'Верно', иначе выведите 'Неверно'. Проверьте работу скрипта при **test**, равном true, false. Напишите два варианта скрипта - с короткой записью и с длинной.
2. Если переменная **test** не равна true, то выведите 'Верно', иначе выведите 'Неверно'. Проверьте работу скрипта при **test**, равном true, false. Напишите два варианта скрипта - с короткой записью и с длинной.

### Работа с && (и) и || (или)

1. Если переменная **a** больше нуля и меньше 5-ти, то выведите 'Верно', иначе выведите 'Неверно'. Проверьте работу скрипта при **a**, равном 5, 0, -3, 2.
2. Если переменная **a** равна нулю или равна двум, то прибавьте к ней 7, иначе поделите ее на 10. Выведите новое значение переменной на экран. Проверьте работу скрипта при **a**, равном 5, 0, -3, 2.
3. Если переменная **a** равна или меньше 1, а переменная **b** больше или равна 3, то выведите сумму этих переменных, иначе выведите их разность (результат вычитания). Проверьте работу скрипта при **a** и **b**, равном 1 и 3, 0 и 6, 3 и 5.
4. Если переменная **a** больше 2-х и меньше 11-ти, или переменная **b** больше или равна 6-ти и меньше 14-ти, то выведите 'Верно', в противном случае выведите 'Неверно'.

### На switch-case

1. Переменная **num** может принимать **4** значения: 1, 2, 3 или 4. Если она имеет значение **'1'**, то в переменную **result** запишем 'зима', если имеет значение **'2'** – 'весна' и так далее. Решите задачу через **switch-case**.

### Задачи

1. В переменной **day** лежит какое-то число из интервала от **1** до **31**. Определите в какую декаду месяца попадает это число (в первую, вторую или третью).
2. В переменной **month** лежит какое-то число из интервала от **1** до **12**. Определите в какую пору года попадает этот месяц (зима, лето, весна, осень).
3. Дана строка, состоящая из символов, например, 'abcde'. Проверьте, что первым символом этой строки является буква 'a'. Если это так - выведите 'да', в противном случае выведите 'нет'.
4. Дана строка с цифрами, например, '12345'. Проверьте, что первым символом этой строки является цифра 1, 2 или 3. Если это так - выведите 'да', в противном случае выведите 'нет'.
5. Дана строка из 3-х цифр. Найдите сумму этих цифр. То есть сложите как числа первый символ строки, второй и третий.
6. Дана строка из 6-ти цифр. Проверьте, что сумма первых трех цифр равняется сумме вторых трех цифр. Если это так - выведите 'да', в противном случае выведите 'нет'.

**Контрольные вопросы**

1. Для чего нужны операторы ветвления?
2. Назовите разницу между == и ===.
3. Назовите разницу между != и !==.
4. В чем разница между операторами **if-else** и **swich-case**?