

УСЛОВИЯ



Вопросы для повторения материала:

Как объявить
переменную?



Вопросы для повторения материала:

Как объявить
переменную?



Имя переменной =
значение





Вопросы для повторения материала:

Как объявить
переменную?



Имя переменной =
значение



Как вывести на экран
информацию?



Вопросы для повторения материала:

Как объявить
переменную?



Имя переменной =
значение



Как вывести на экран
информацию?



Функцией print().





Вопросы для повторения материала:

Как объявить переменную?



Имя переменной = значение



Как вывести на экран информацию?



Функцией print().



Как можно ввести информацию?



Вопросы для повторения материала:

Как объявить переменную?



Имя переменной = значение



Как вывести на экран информацию?



Функцией print().



Как можно ввести информацию?



Функцией input()





Вопросы для повторения материала:

Как объявить переменную?



Имя переменной = значение



Как вывести на экран информацию?



Функцией print().



Как можно ввести информацию?



Функцией input()



Какие типы данных мы изучили?



Вопросы для повторения материала:

Как объявить переменную?



Имя переменной = значение



Как вывести на экран информацию?



Функцией print().



Как можно ввести информацию?



Функцией input()



Какие типы данных мы изучили?



int, str, float, boolean





Вопросы для повторения материала:

Как объявить переменную?



Имя переменной = значение



Как вывести на экран информацию?



Функцией print().



Как можно ввести информацию?



Функцией input()



Какие типы данных мы изучили?



int, str, float, boolean



Как превратить строку в число?



Вопросы для повторения материала:

Как объявить переменную?



Имя переменной = значение



Как вывести на экран информацию?



Функцией print().



Как можно ввести информацию?



Функцией input()



Какие типы данных мы изучили?



int, str, float, boolean

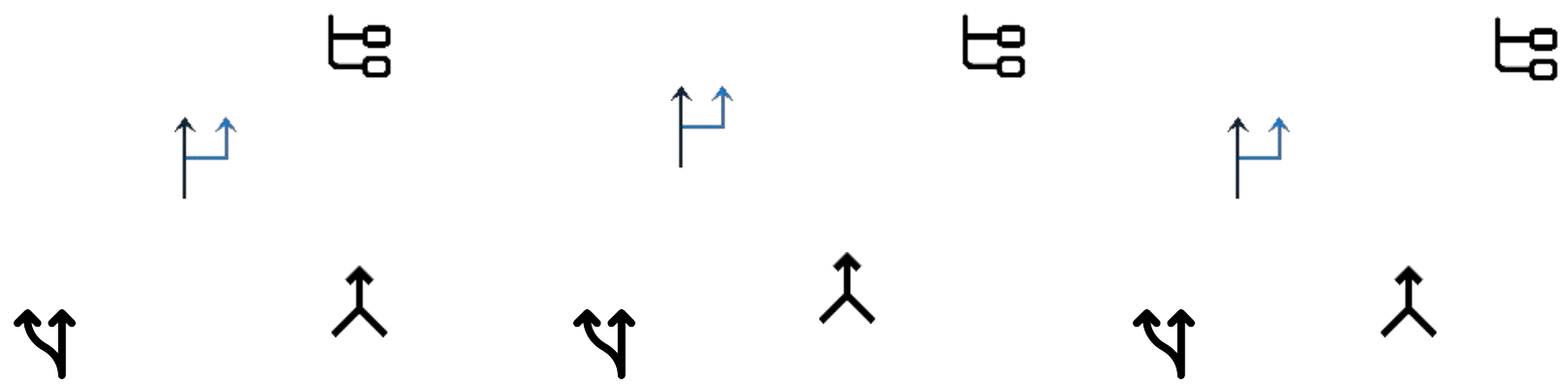


Как превратить строку в число?

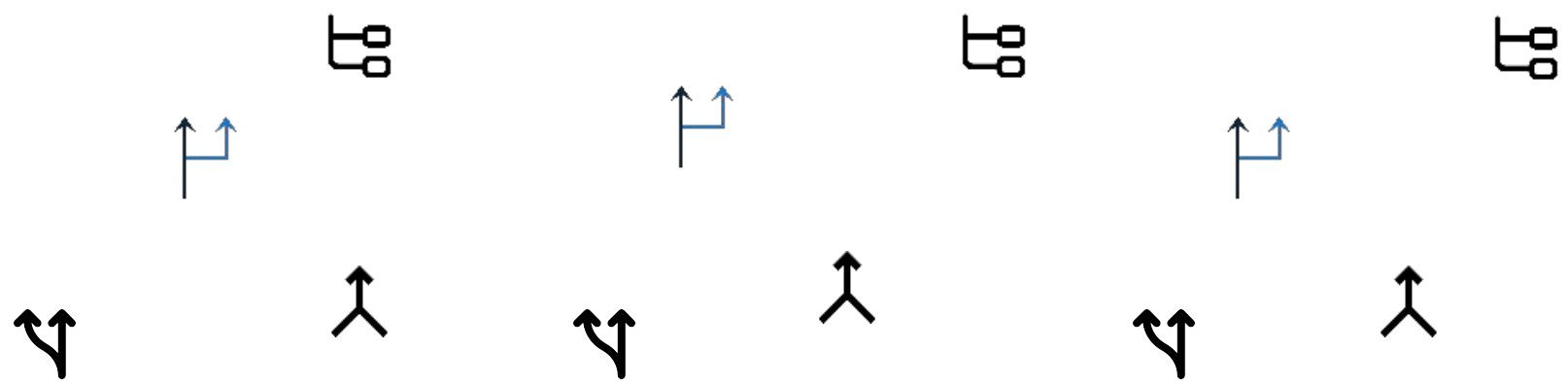


Функцией int()



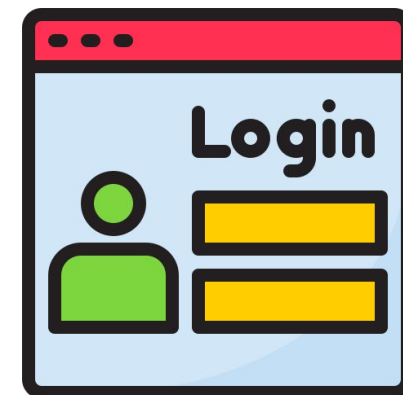


Ветвление в программах. Условия

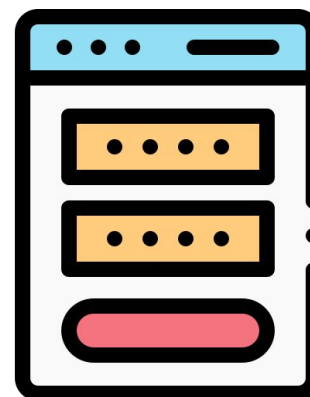
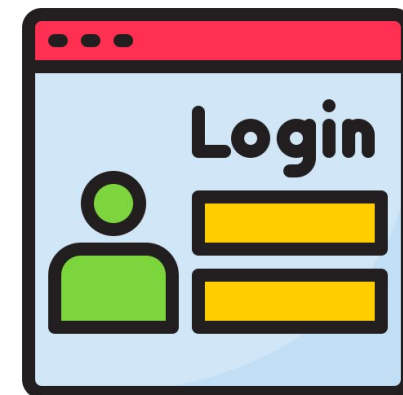




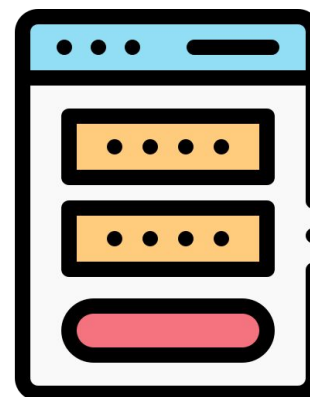
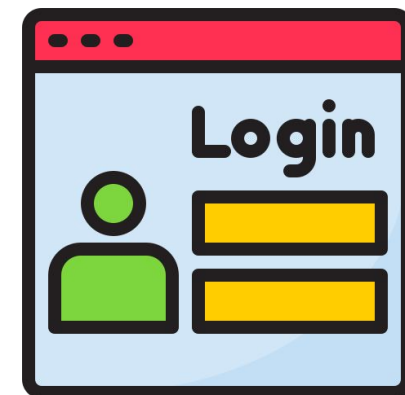
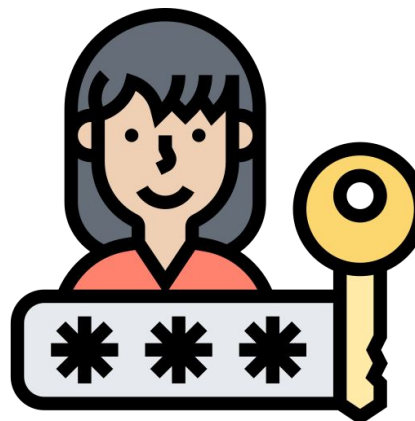
Введите пин-
код!



Логин и пароль!



Войдите в
аккаунт!



Доступ
ограничен!



Авторизация

Обязательное поле

Обязательное поле

Войти

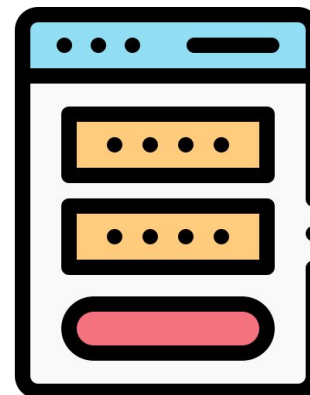
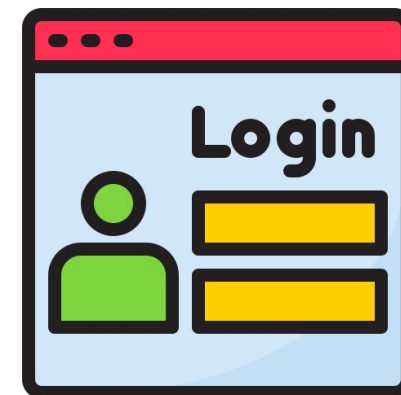
Вход для участников «Код будущего»

20.35 УНИВЕРСИТЕТ

Войти через сайт educont.ru

Забыли пароль? [Сбросить пароль](#)

Новый пользователь? [Зарегистрироваться](#)





Авторизация


Обязательное поле

Обязательное поле

Войти

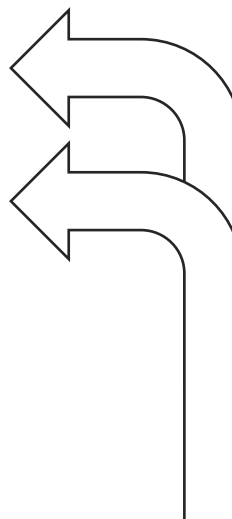
Вход для участников «Код будущего»

20.35 УНИВЕРСИТЕТ

 Войти через сайт educont.ru

Забыли пароль? [Сбросить пароль](#)

Новый пользователь? [Зарегистрироваться](#)



Верный и неверный ввод - **разные реакции!**



Авторизация


Обязательное поле

Обязательное поле

Войти

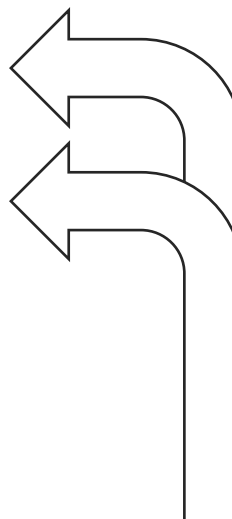
Вход для участников «Код будущего»

20.35 УНИВЕРСИТЕТ

 Войти через сайт educont.ru

Забыли пароль? [Сбросить пароль](#)

Новый пользователь? [Зарегистрироваться](#)



Ввели **верно** - нас пускают в модуль.

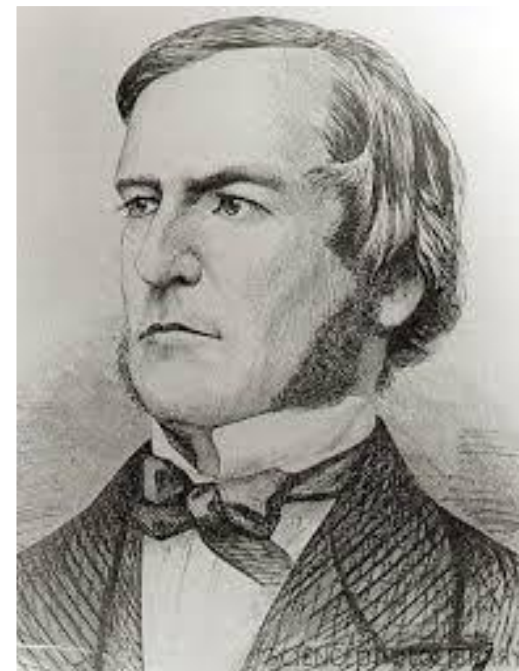
Ввели **неверно** - в модуль не пускают, полноценно поработать не получится.

-> Имеем **ветвление!**

Верный и неверный ввод - **разные реакции!**

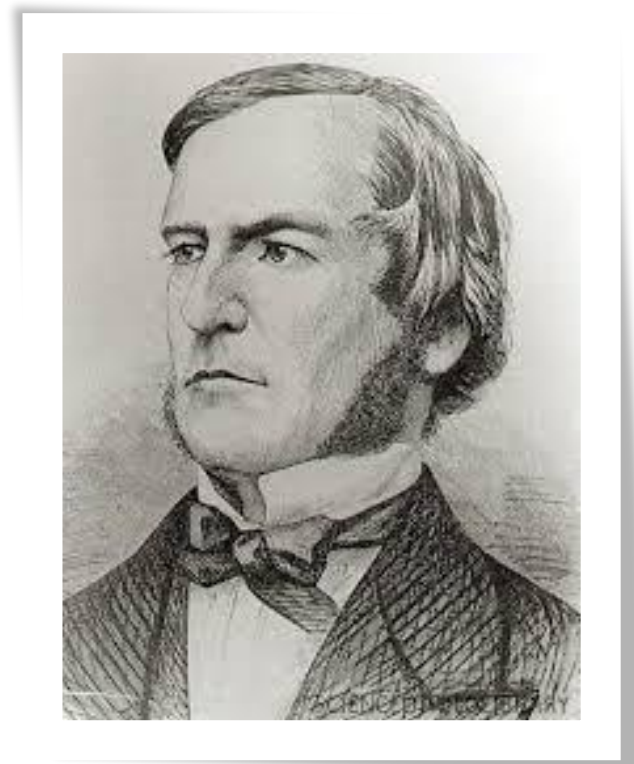
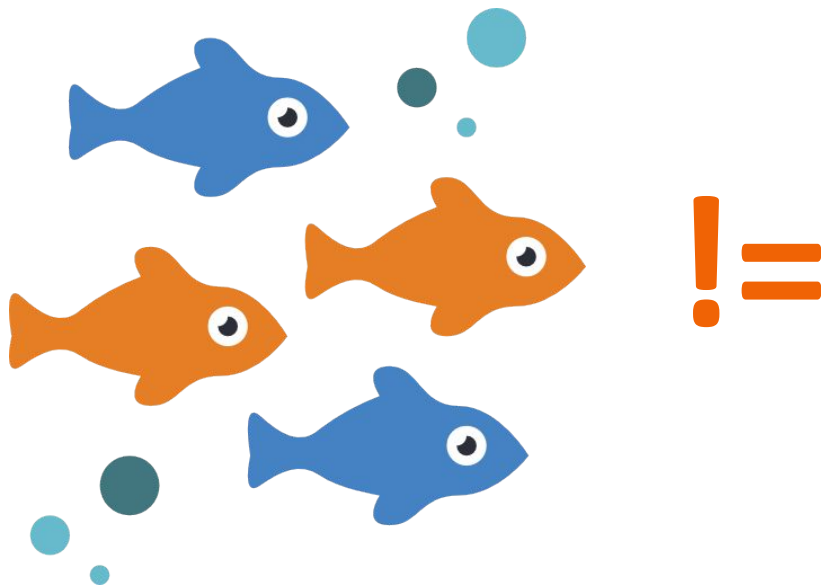


Джордж Буль— английский ученый, отец математической (булевой) логики.





Джордж Буль— английский ученый, отец математической (булевой) логики.





Порассуждаем:



Порассуждаем:

Сегодня у нас есть
занятие?



Порассуждаем:

Сегодня у нас есть
занятие?



Да!



Порассуждаем:

Сегодня у нас есть
занятие?



Да!



читаем теорию
подключаемся к занятию
работаем



Порассуждаем:

Сегодня у нас есть занятие?



Да!



читаем теорию
подключаемся к занятию
работаем

Завтра у нас есть занятие?



Порассуждаем:

Сегодня у нас есть занятие?



Да!



читаем теорию
подключаемся к занятию
работаем

Завтра у нас есть занятие?



Нет!



Порассуждаем:

Сегодня у нас есть занятие?



Да!



читаем теорию
подключаемся к занятию
работаем

Завтра у нас есть занятие?



Нет!



не читаем теорию
не подключаемся к занятию
не работаем



Порассуждаем:

Сегодня у нас есть занятие?



Да!



читаем теорию
подключаемся к занятию
работаем

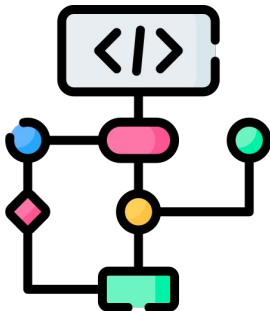
Завтра у нас есть занятие?



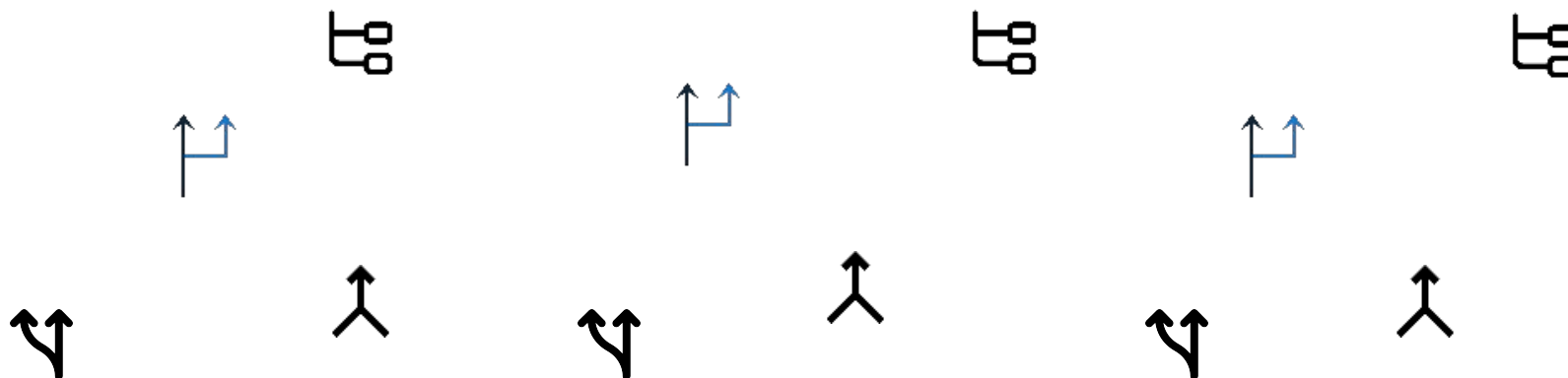
Нет!



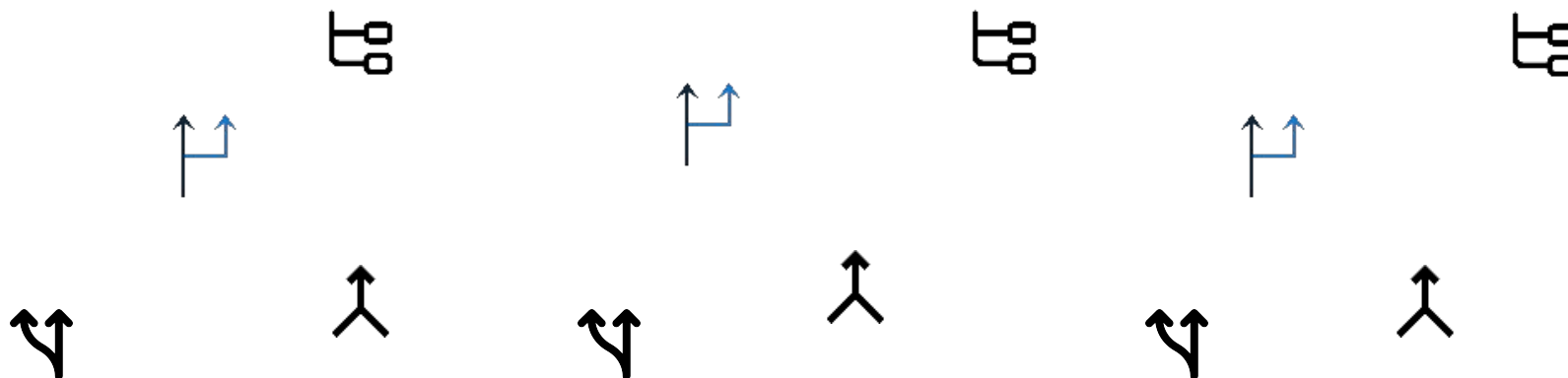
не читаем теорию
не подключаемся к занятию
не работаем



Нелинейность!



Условный оператор. Логические операторы





Условный оператор —



Условный оператор — это конструкция языка программирования, определяющая более одной последовательности выполнения в программе.



Условный оператор — это конструкция языка программирования, определяющая более одной последовательности выполнения в программе.



В Python – **if** (если).

if (какое-то условие):
 (что делается, если оно выполняется)



Оператор сравнения —



Оператор сравнения — знак или сочетание знаков, задающих форму условия.



Оператор сравнения — знак или сочетание знаков, задающих форму условия.

!= (не равно)



Оператор сравнения — знак или сочетание знаков, задающих форму условия.

!= (не равно)

== (равно)



Оператор сравнения — знак или сочетание знаков, задающих форму условия.

!= (не равно)

== (равно)

>= (больше или равно)



Оператор сравнения — знак или сочетание знаков, задающих форму условия.

!= (не равно)

== (равно)

>= (больше или равно)

<= (меньше или равно)



Оператор сравнения — знак или сочетание знаков, задающих форму условия.

!= (не равно)

== (равно)

>= (больше или равно)

<= (меньше или равно)

> (строго больше)



Оператор сравнения — знак или сочетание знаков, задающих форму условия.

\neq (не равно)

$=$ (равно)

\geq (больше или равно)

\leq (меньше или равно)

$>$ (строго больше)

$<$ (строго меньше)



Если мы сравниваем **текстовую** информацию, то мы можем:



Если мы сравниваем **текстовую** информацию, то мы можем:

Проверить её на равенство (==).





Если мы сравниваем **текстовую** информацию, то мы можем:

Проверить её на равенство (==).



Проверить её на неравенство (!=).





Если мы сравниваем **текстовую** информацию, то мы можем:

Проверить её на равенство (==).



Проверить её на неравенство (!=).



1010001101011
1010110101000
10100011
10100011



А если мы будем сравнивать **числовую** информацию, то мы можем:



Если мы сравниваем **текстовую** информацию, то мы можем:

Проверить её на равенство (==).



Проверить её на неравенство (!=).



1010001101011
1010110101000
10100011
10100011



А если мы будем сравнивать **числовую** информацию, то мы можем:

Проверить, больше (>) число или меньше (<) другого.





Если мы сравниваем **текстовую** информацию, то мы можем:

Проверить её на равенство (==).



Проверить её на неравенство (!=).



1010001101011
1010110101000
10100011
10100011



А если мы будем сравнивать **числовую** информацию, то мы можем:

Проверить, больше (>) число или меньше (<) другого.



Проверить, больше либо равно (>=) или меньше либо равно (<=) число в сравнении с другим.





Какой оператор нам понадобится для следующих условий?

| | |
|--|--|
| | |
| | |
| | |
| | |
| | |



Какой оператор нам понадобится для следующих условий?

"Пол сотрудника - не мужской"





Какой оператор нам понадобится для следующих условий?

"Пол сотрудника - не мужской"

не равно (!=)



Какой оператор нам понадобится для следующих условий?

"Пол сотрудника - не мужской"

не равно (!=)

"Балл за экзамен - как минимум 80"



Какой оператор нам понадобится для следующих условий?

"Пол сотрудника - не мужской"

не равно (!=)

"Балл за экзамен - как минимум 80"

больше или равно (>=)



Какой оператор нам понадобится для следующих условий?

"Пол сотрудника - не мужской"

не равно (!=)

"Балл за экзамен - как минимум 80"

больше или равно (>=)

"Зрители старше двенадцати"



Какой оператор нам понадобится для следующих условий?

"Пол сотрудника - не мужской"

не равно (!=)

"Балл за экзамен - как минимум 80"

больше или равно (>=)

"Зрители старше двенадцати"

больше или равно



Какой оператор нам понадобится для следующих условий?

"Пол сотрудника - не мужской"

не равно (!=)

"Балл за экзамен - как минимум 80"

больше или равно (>=)

"Зрители старше двенадцати"

больше или равно

"Месяц - июнь"



Какой оператор нам понадобится для следующих условий?

"Пол сотрудника - не мужской"

не равно (!=)

"Балл за экзамен - как минимум 80"

больше или равно (>=)

"Зрители старше двенадцати"

больше или равно

"Месяц - июнь"

равно (==)



Какой оператор нам понадобится для следующих условий?

"Пол сотрудника - не мужской"

не равно (!=)

"Балл за экзамен - как минимум 80"

больше или равно (>=)

"Зрители старше двенадцати"

больше или равно

"Месяц - июнь"

равно (==)

"Дни, начиная с завтрашнего"



Какой оператор нам понадобится для следующих условий?

"Пол сотрудника - не мужской"

не равно (!=)

"Балл за экзамен - как минимум 80"

больше или равно (>=)

"Зрители старше двенадцати"

больше или равно

"Месяц - июнь"

равно (==)

"Дни, начиная с завтрашнего"

больше (>) / больше или равно



Какой оператор нам понадобится для следующих условий?

"Пол сотрудника - не мужской"

не равно (!=)

"Балл за экзамен - как минимум 80"

больше или равно (>=)

"Зрители старше двенадцати"

больше или равно

"Месяц - июнь"

равно (==)

"Дни, начиная с завтрашнего"

больше (>) / больше или равно





У нас есть несколько чисел, требуется их сравнить и вывести самое большое и самое маленькое из них.



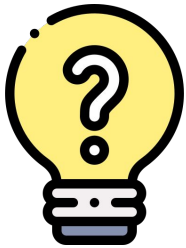
У нас есть несколько чисел, требуется их сравнить и вывести самое большое и самое маленькое из них.



- Можно реализовать это через условие?



У нас есть несколько чисел, требуется их сравнить и вывести самое большое и самое маленькое из них.



- Можно реализовать это через условие?
- Да, но чем больше чисел, тем более громоздким оно будет получаться:



У нас есть несколько чисел, требуется их сравнить и вывести самое большое и самое маленькое из них.



- Можно реализовать это через условие?
- Да, но чем больше чисел, тем более громоздким оно будет получаться:

`if number_1 > number_2 and number_1 > number_3 and number_1 > number_4...`



У нас есть несколько чисел, требуется их сравнить и вывести самое большое и самое маленькое из них.



- Можно реализовать это через условие?
- Да, но чем больше чисел, тем более громоздким оно будет получаться:

`if number_1 > number_2 and number_1 > number_3 and number_1 > number_4...`



Проверили только одно число, и только на превосходство!



У нас есть несколько чисел, требуется их сравнить и вывести самое большое и самое маленькое из них.



- Можно реализовать это через условие?
- Да, но чем больше чисел, тем более громоздким оно будет получаться:

`if number_1 > number_2 and number_1 > number_3 and number_1 > number_4...`

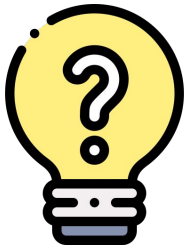


Проверили только одно число, и только на превосходство!

Две функции для облегчения жизни:



У нас есть несколько чисел, требуется их сравнить и вывести самое большое и самое маленькое из них.



- Можно реализовать это через условие?
- Да, но чем больше чисел, тем более громоздким оно будет получаться:

```
if number_1 > number_2 and number_1 > number_3 and number_1 > number_4...
```



Проверили только одно число, и только на превосходство!

Две функции для облегчения жизни:

`max()` - принимает набор значений и находит самое большое



У нас есть несколько чисел, требуется их сравнить и вывести самое большое и самое маленькое из них.



- Можно реализовать это через условие?
- Да, но чем больше чисел, тем более громоздким оно будет получаться:

```
if number_1 > number_2 and number_1 > number_3 and number_1 > number_4...
```



Проверили только одно число, и только на превосходство!

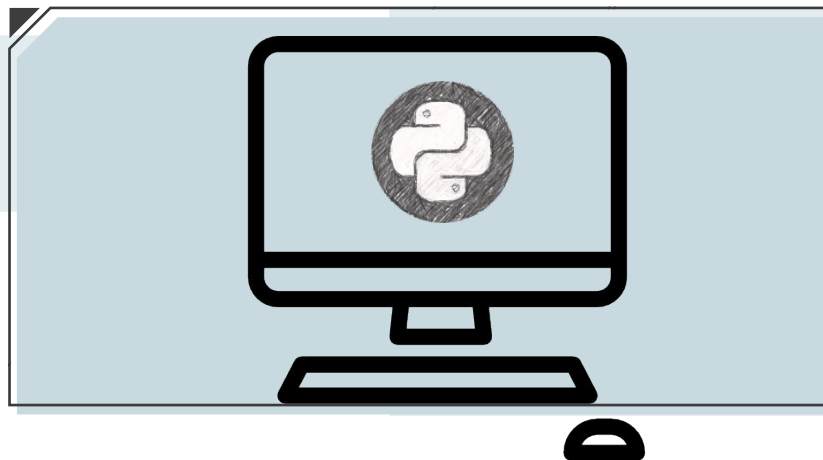
Две функции для облегчения жизни:

`max()` - принимает набор значений и находит самое большое

`min()` - принимает набор значений и находит самое маленькое



Проверка
понимания.

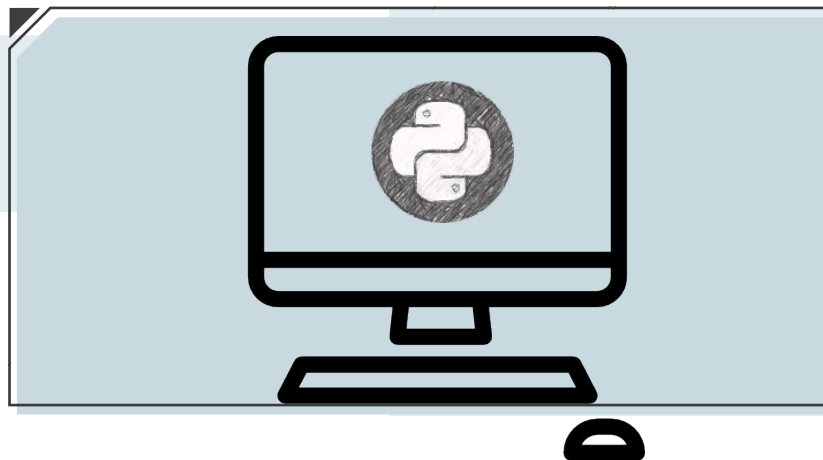


1010001101011
1010110101000
10100011
10100011





Проверка понимания.



1010001101011
1010110101000
10100011
10100011

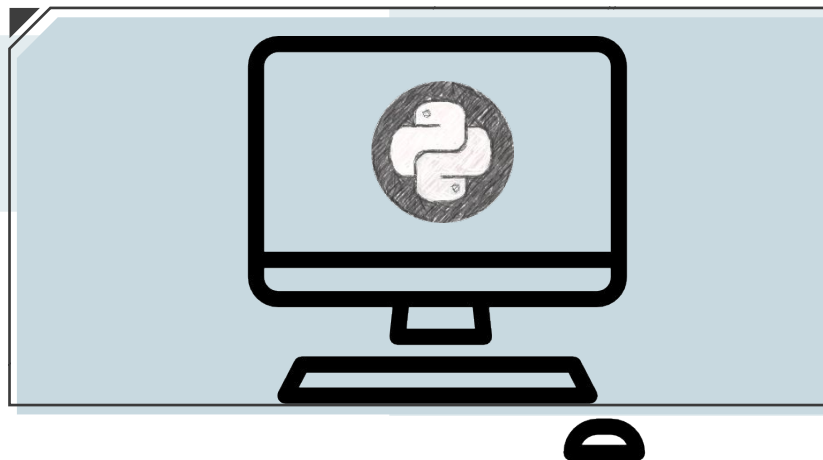


1

Каким условным оператором можно добавлять условия, если их несколько?



Проверка
понимания.



1010001101011
1010110101000
10100011
10100011



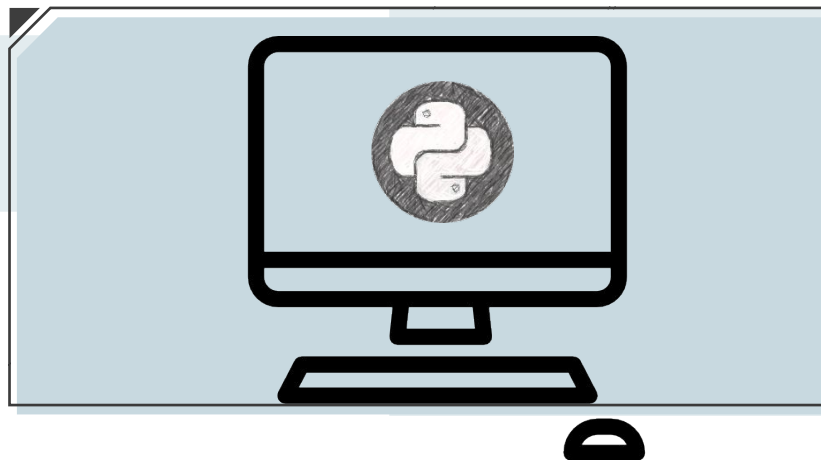
1

Каким условным оператором можно добавлять условия, если их несколько?

elif



Проверка понимания.



1010001101011
1010110101000
10100011
10100011



1

Каким условным оператором можно добавлять условия, если их несколько?

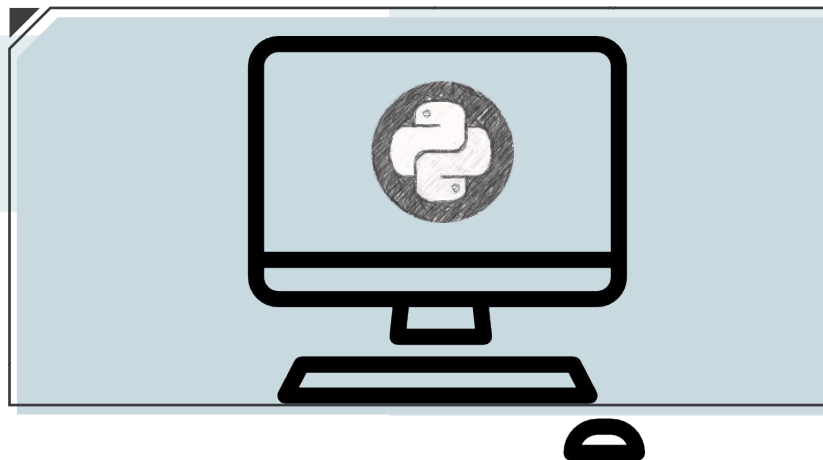
elif

2

Что значит else?



Проверка понимания.



1010001101011
1010110101000
10100011
10100011



1 Каким условным оператором можно добавлять условия, если их несколько?

elif

2 Что значит else?

Иначе



Проверка понимания.



1010001101011
1010110101000
101000111
101000111



1 Каким условным оператором можно добавлять условия, если их несколько?

elif

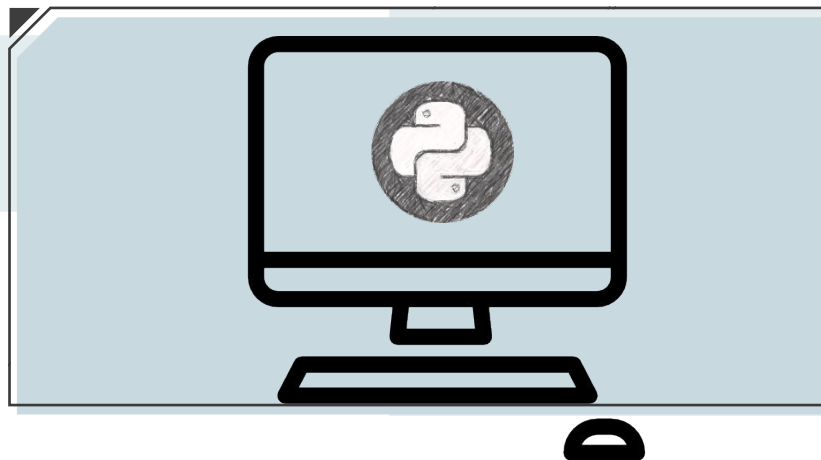
2 Что значит else?

Иначе

3 Как проверить равенство в условии?



Проверка понимания.



1010001101011
1010110101000
101000111
10100011



1 Каким условным оператором можно добавлять условия, если их несколько?

elif

2 Что значит else?

Иначе

3 Как проверить равенство в условии?

==



Проверка понимания.



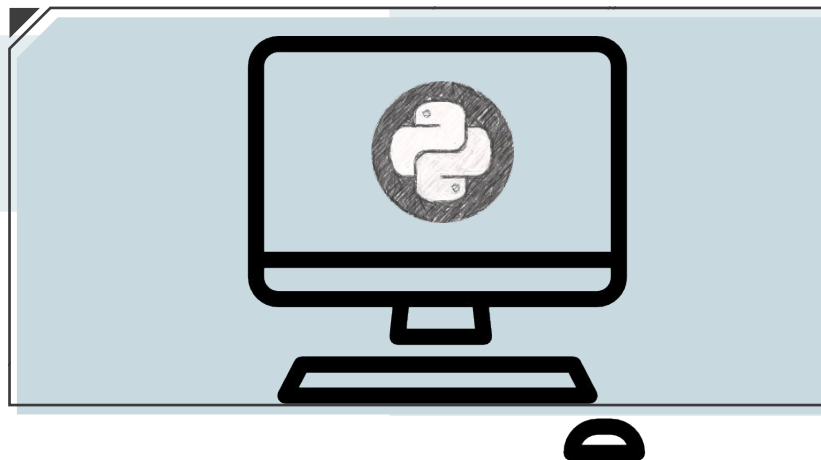
1010001101011
1010110101000
101000111
101000111



- | | | |
|---|--|-------|
| 1 | Каким условным оператором можно добавлять условия, если их несколько? | elif |
| 2 | Что значит else? | Иначе |
| 3 | Как проверить равенство в условии? | == |
| 4 | Какой оператор проверяет истинность хотя бы одной из частей логического выражения? | |



Проверка понимания.



1010001101011
1010110101000
10100011
10100011



1 Каким условным оператором можно добавлять условия, если их несколько?

elif

2 Что значит else?

Иначе

3 Как проверить равенство в условии?

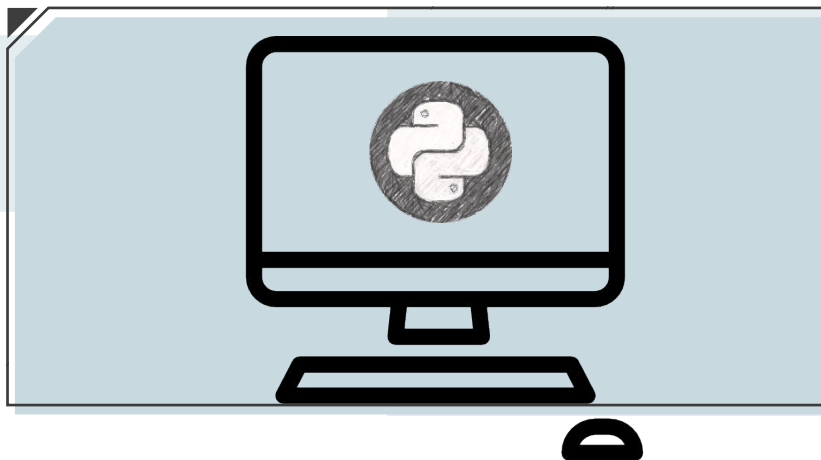
==

4 Какой оператор проверяет истинность хотя бы одной из частей логического выражения?

or (или)



Проверка понимания.



1010001101011
1010110101000
101000111
101000111



1 Каким условным оператором можно добавлять условия, если их несколько?

elif

2 Что значит else?

Иначе

3 Как проверить равенство в условии?

==

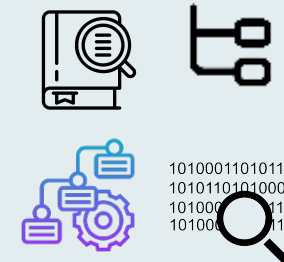
4 Какой оператор проверяет истинность хотя бы одной из частей логического выражения?

or (или)

5 Сколько операторов elif может быть у одного условия?



Проверка понимания.



- | | | |
|---|--|----------------|
| 1 | Каким условным оператором можно добавлять условия, если их несколько? | elif |
| 2 | Что значит else? | Иначе |
| 3 | Как проверить равенство в условии? | == |
| 4 | Какой оператор проверяет истинность хотя бы одной из частей логического выражения? | or (или) |
| 5 | Сколько операторов elif может быть у одного условия? | Сколько угодно |



Подпишись на наш
канал

