Kypc Java



Карта квестов Список лекций CS50 Android

Ввод с клавиатуры

Java Syntax 3 уровень, 7 лекция

ОТКРЫТА

- Амиго, настало твоё время теперь я расскажу про ввод с клавиатуры.
- Для вывода данных на экран мы использовали **System.out**. Для ввода данных будем использовать **System.in**.
- Звучит не сложно.
- Но у **System.in** есть минус он позволяет считать с клавиатуры только коды символов. Чтобы обойти эту проблему и считывать большие порции данных за один раз, мы будем использовать более сложную конструкцию:

```
Пример 1
Ввод строки и числа с клавиатуры
```

```
1 InputStream inputStream = System.in;
2 Reader inputStreamReader = new InputStreamReader(inputStream);
3 BufferedReader bufferedReader = new BufferedReader(inputStreamReader);
4
5 String name = bufferedReader.readLine(); //читаем строку с клавиатуры
6 String sAge = bufferedReader.readLine(); //читаем строку с клавиатуры
7 int nAge = Integer.parseInt(sAge); //преобразовываем строку в число.
```



Набираем код

Иногда думать не надо, строчить надо! Как ни парадоксально звучит, порой пальцы «запоминают» лучше, чем сознание. Вот почему во время обучения в секретном центре JavaRush вы иногда встречаете задания на набор кода. Набирая код, вы привыкаете к синтаксису и зарабатываете немного материи. А ещё — боретесь с ленью.

Открыть

Пример 2 Более компактная запись первой части:

```
BufferedReader reader = new BufferedReader(new InputStreamReader(Syste

String name = reader.readLine();

String sAge = reader.readLine();

int nAge = Integer.parseInt(sAge);
```

Пример 3

Еще более компактная

```
1 Scanner scanner = new Scanner(System.in);
2 String name = scanner.nextLine();
3 int age = scanner.nextInt();
```

- Вопросы?
- Эээ... Ничего не понятно.
- Чтобы считать строку с клавиатуры, удобнее всего воспользоваться объектом **BufferedReader**. Но для этого в него нужно передать объект, из которого он будет вычитывать данные **System.in**.
- Ho **System.in** и **BufferedReader** не совместимы друг с другом, поэтому мы используем переходник еще один объект **InputStreamReader**.
- Вроде ясно. А что еще за класс Scanner?
- Использовать Scanner довольно удобно, но от этого не очень много пользы. Дело в том, что в будущем (и в учебе, и на работе) ты будешь часто использовать объекты BufferedReader и

InputStreamReader и очень-очень редко объект типа Scanner. В данной ситуации он удобен, но в будущем толку от него мало. Так что мы пользоваться им не будем.

— Что-то вроде понятно, но не уверен, что все и правильно.

< Предыдущая





Комментарии (168)

популярные новые старые

Vitaly Morochenets

Введите текст комментария

funkagenda 3 уровень

суббота, 20:46

•••

Как я понял, но на более простом языке по примеру 2. Поправьте если где-то не прав:

Строка 1. System.in - читает только байты, а это нам не нужно. Когда мы введем символ(буквы, слова) будет выведен на экран этот символ в байтах. Для того, что бы выводить на экран символы, передаем System.in в класс InputStreamReader и получается new InputStreamReader(System.in). Что бы читать строки, передаем InputStreamReader(System.in) в класс BufferedReader, получается new BufferedReader(new InputStreamReader(System.in)).

Строка 3. Создаем объект name, читаем вводимые с клавиатуры символы

Строка 4. Создаем объект sAge, читаем вводимые с клавиатуры символы.

Строка 5. Т.к. нам нужен возраст в цифрах, то для этого используем метод Integer.parseInt (преобразование строки в число).

После запуска программы сначала нужно ввести символы, а замет цифры, иначе будет ошибка.

Ответить



Денис Сава 5 уровень, Москва

3 ноября, 16:17

•••

Надеюсь дальше будет больше объяснений.

Хорошо что пользуюсь еще другими ресурсами. А то бы завис на этом месте.

Впрочем как можно писать такие вещи еще не объяснив абстрактные классы.

Хотя с другой стороны на сторонних ресурсах тоже начинают со сканера не объясняя его.

Ладно пойдем дальше.

Ответить

