



Жесткая типизация

Java Syntax (/quests/QUEST_JAVA_SYNTAX)

10 уровень (/quests/lectures/?quest=QUEST_JAVA_SYNTAX&level=10), 1 лекция (/quests/lectures/questsyntax.level10.lecture01)

ОТКРЫТА

— Привет, Амиго!

— Привет, Элли!

— Сегодня у меня хорошее настроение, поэтому расскажу тебе кое-что интересное. И начну я с типизации примитивных типов в Java.

— В Java у каждого объекта и у каждой переменной есть свой жёстко заданный неизменяемый тип. Тип переменной определяется ещё в процессе компиляции программы, тип объекта – при его создании. Тип нового созданного объекта и/или переменной остаются неизменными все их время жизни. Пример:

Код на Java

Описание

```
1 int a = 11;
2 int b = 5;
3 int c = a / b; // c == 2
```

`a / b` – это деление нацело. Ответом будет два. Остаток от деления просто отбрасывается.

```
1 int a = 13;
2 int b = 5;
3 int d = a % b; // d == 3
```

В `d` будет храниться остаток от деления `a` на `b` нацело. Остаток равен 3.

— Но есть и пара интересных нюансов, которые следует помнить.

— Во-первых, ссылочная переменная не всегда хранит значение такого же типа, как и она.

— Во-вторых, при взаимодействии переменных двух разных типов, они должны быть сначала преобразованы к одному общему типу.

— А как же деление? Если разделить 1 на 3, мы получим 0.333(3). Ведь так?

— Нет не так. При делении двух целых чисел, результатом тоже будет целое число. Если разделить 5 на 3, то ответом будет 1 и два в остатке. Остаток при этом отбрасывается.

— Если разделить 1 на 3, то мы получим 0 (и единицу в остатке, который отбросится).

— А что делать, если я все же хочу получить 0.333 ?

— В Java числа перед делением лучше всего приводить к вещественному (дробному) типу путем умножения их на вещественную единицу – 1.0

Код на Java

Описание

```
1 int a = 1/3;
```

`a` будет содержать 0

```
1 double d = 1/3;
```

`d` будет содержать 0.0

```
1 double d = 1.0 / 3;
```

`d` будет содержать 0.333(3)

```
1 double d = 1 / 3.0;
```

`d` будет содержать 0.333(3)

```
1 int a=5, b=7;  
2 double d = (a*1.0) / b;
```

`d` будет содержать 0.7142857142857143

— Понятно.

[< \(/quests/lectures/questsyntax.level10.lecture00\)](/quests/lectures/questsyntax.level10.lecture00)

[> \(/quests/lectures/questsyntax.level10.lecture02\)](/quests/lectures/questsyntax.level10.lecture02)



0



0



0



0



0

+19

Комментарии (20)

популярные

новые

старые

Никита

Введите текст комментария

Fs Jt 20 уровень, Киев

12 декабря 2017, 23:07



а не проще ли в последнем примере double d = (double)a/b?

Ответить

+4

Дмитрий Оносовский 11 уровень, Одесса

27 февраля, 20:08



тоже подумал, а почему бы не привести явно к нужному типу...

Ответить

0

Vikentsi 22 уровень, Минск

24 июня 2017, 18:48



Вы тут постоянно ворчите, где и что должно быть на первом месте.

А теперь представьте если бы вы родились и родители с появлением первых ваших звуков посадили бы вас за парту в месячном возрасте и начали рассказывать историю языка. Потом сразу типы все животных, предметов и т.д. Вместе с этим бы акцентировали ваше внимание на грамматике. Я думаю вы бы сошли сумма или возненавидели своих родителей.

Вместо этого они с вами просто говорят, акцентируют иногда ваше внимание на основных словах, цветах. НЕ ПЕРЕГРУЖАЮТ ВАС ИНФОРМАЦИЕЙ. И потом вы вуаля уже говорите на родном языке, а потом в первых классах вуаля и уже пишете и читаете и вам кажется так элементарно. Так что рты по закрывали и занимайтесь.

Ответить

+36

Akira Rokudo 20 уровень, Москва

24 июня 2017, 18:12



"Во-первых, ссылочная переменная не всегда хранит значение такого же типа, как и она."

Для тех кто не понял. Это в основном отсылка к наследованию. Ничто не мешает написать следующий код.(есть класс A и B, B наследник класса A)

A a = new B();

Ссылочная переменная типа A, а хранит она значение типа B.

Мы конечно можем написать затем a = (B)a; Но все равно придется для обращения к методам B каждый раз писать ((B) a).methodB(); т.е. производить сужение.

Ответить

+2

Артем Дяченко 11 уровень, Киев

6 августа 2017, 17:04



A a = new B(); - это чистой воды полиморфизм

Ответить

+2

Harvey 27 уровень, Минск

3 сентября 2017, 19:15



По сути наследование ведет к полиморфизму. Полиморфизм является побочным преимуществом наследования. Поэтому сложно разделить эти 2 понятия, т.к. они работают вместе.

Ответить

+1

Akira Rokudo 20 уровень, Москва

3 сентября 2017, 20:46



ну если извратиться, то да, классы реализующие интерфейсы, являются их наследниками) и именно такая реализация полиморфизма наиболее часто встречается)

Ответить

0

Даниил Быстров 16 уровень

9 мая 2017, 19:17



очепятка Если разделить 1 на 3, то мы получим 0 (и единицу в остатке, который отбросится).

Ответить

+2

Евгений Герцык 10 уровень, Одесса

23 октября 2017, 10:17



По моему, все верно, остаток, который отбросится.

Ответить

0

Александр Дубяга 16 уровень

6 апреля 2017, 08:11



Эта лекция на своем месте!!!в проивном случае мы бы вообще ничего сами не искали и не думали

0

Ответить

zulan 11 уровень, Bangkok

29 марта 2017, 11:58

...

всем кому не понятно было это вначале и долго разбирались, теперь все очень понятно, и материал уже ложится в основу знаний по ява, в этом смысл такой подачи материала, тут так будет до самого конца

0

Ответить

Harvey 27 уровень, Минск

3 сентября 2017, 19:17

...

Наверное все таки правильной будет джава. При просмотре видеороликов на английском, язык называют джавой.

0

Ответить

Юрій Якимчук 16 уровень

27 сентября 2017, 12:41

...

потому что Ява - это ж мотоцикл! :)

+5

Ответить

Александр Клименко 34 уровень, Днепр

27 марта 2017, 09:46

...

Согласен с комментариями выше - это лекция должна быть одной из первых. Не совсем понятна методика предоставления материала :(

+2

Ответить

Anonymous #469834 11 уровень, Самара

3 марта 2017, 02:26

...

Зачем умножать на вещественную единицу, если можно просто привести тип?

0

Ответить

Alexey Smirnov 29 уровень, Москва

8 апреля 2017, 13:39

...

Вот да.

0

Ответить

Илья Андреев 27 уровень, Самара

16 февраля 2017, 19:54

...

Зеленая лягушка, про саму ссылочную переменную и речь. Ссылочная переменная не всегда хранит значение того же типа который у неё(ссылочной переменной) самой.

0

Ответить

Эрик Айткулов 34 уровень, Санкт-Петербург

15 февраля 2017, 15:12

...

эта тема очень нужна в восьмом уровне. здесь она "повторение - мать учения". при решении задач пользовался приведением типов:
int a=5, b=7;
double d = (double) a / b;

+2

Ответить

[Загрузить еще](#)

[JavaRush.ru](#) [G+ \(https://plus.google.com/114772402300089087607\)](#) [Twitter \(https://twitter.com/javarush_ru\)](#) [Facebook](#)



Программистами не рождаются
© 2018