

Casting

Do Now

What is the output of the following code?

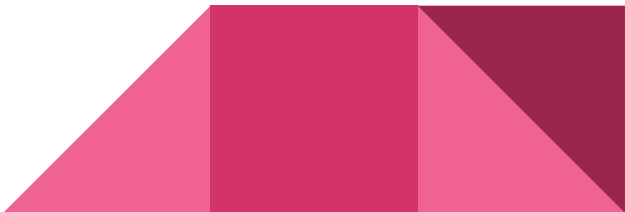
```
int x = 5;
```

```
double y = 15.0;
```

```
System.out.println(y / x);
```

Output: 3.0

What can we do to have an int as result?



Casting

Casting is the conversion of one data type into another; for example, from an integer to a string or vice versa.

Casting in Java:

```
int myNum = (int)10.2    - - - ->    10
```



Casting a double to an int

```
double x = 5.3;
```

```
int y = x;
```

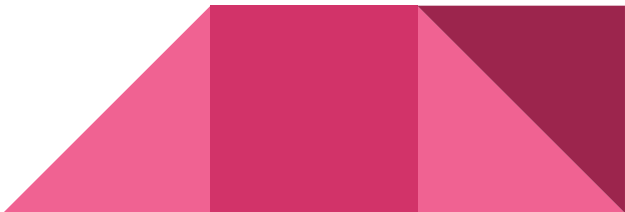
Does it work?

NO. Java will throw an error

How can we make it work?

```
double x = 5.3;
```

```
int y = (int)x;
```



Casting an int to a double

```
int x = 5;
```

```
double y = x;
```

Does it work?

YES. There is a process called **implicit casting** that makes it work. Java automatically casts the value.



Division

```
int money = 100;
```

```
int numPeople = 40;
```

```
double dollarsPerPerson = money / numPeople;
```

How can `money/numPeople` become double without changing the type of variables `money` and `numPeople`?

```
double dollarsPerPerson = (double)money / numPeople;
```

Output: 2.5



Rounding with Casting

Casting can be used to round double values to the closest integer using:

`(int)(x + 0.5)` ---> 0.5 make sure that the value of x will be closest to the correct integer

```
double x = 2.0
```

```
int y = (int)(x + 0.5);
```

```
double x = 2.8
```

```
int y = (int)(x + 0.5);
```



Let's practice

Write a program that:

1. Asks the user for two integers called `int1` and `int2`
2. Divides the two `ints`
3. Prints the result

The trick here is that we want the division of the two `ints` to result in a `double`! Casting values to `doubles` will be necessary to solve this exercise.

Here is a sample run of the solution:

First Int: 3

Second Int: 4

0.75

