

Типы данных, литералы,

Java является языком со строгой типизацией, это значит, что для того, чтоб создать объект мы должны знать, какого типа он будет.

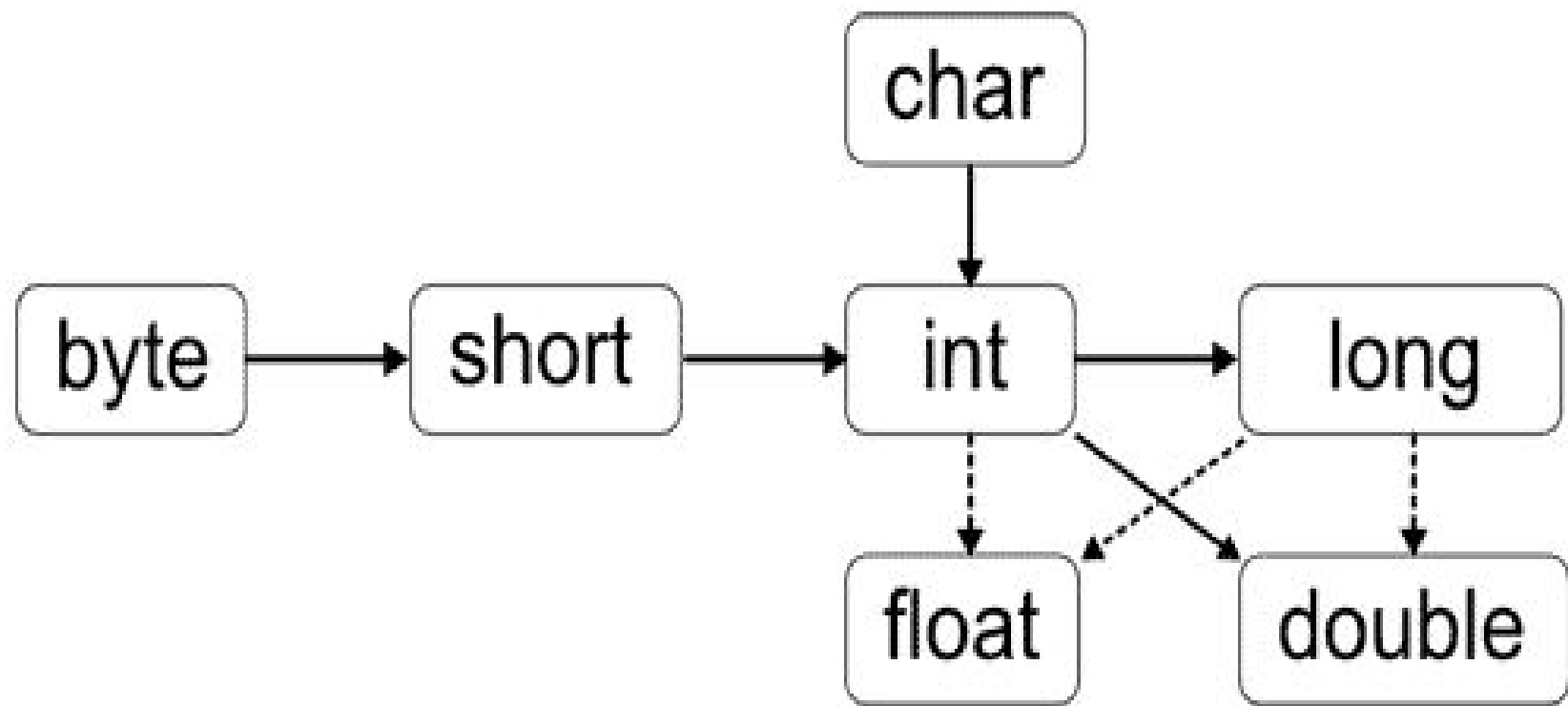
Тип - это классификация, основанная на содержимом и индивидуальности. В программировании тип определяет свойства выражения или переменной таким образом, что можно предсказать ее поведение.

Литералы — это константы, которые записаны по правилам языка Java. Литералы бывают числовые, символьные, булевы и строковые (о строках и строковых литералах речь пойдет на одном из следующих занятий).

Тип данных	Объём занимаемой памяти	Диапазон допустимых значений	Значение по умолчанию (для полей класса)
byte	1 байт	Целые из [-128;127]	0
short	2 байта	Целые из [-32768; 32767]	0
int	4 байта	Целые из [-2147483648; 2147483647]	0
long	8 байт	Целые из [-2 ⁶³ ;2 ⁶³]	0
float	4 байта	Вещественные положительные и отрицательные числа от $\sim 1,4 \cdot 10^{-45}$ до $\sim 3,4 \cdot 10^{38}$ по стандарту IEEE 754	0
double	8 байт	Вещественные положительные и отрицательные числа от $\sim 4,9 \cdot 10^{-324}$ до $\sim 1,8 \cdot 10^{308}$ по стандарту IEEE 754	0
char	2 байта	Натуральные из [0;65535], интерпретируются как коды символов по таблице Unicode	‘\u0000’
boolean	Для хранения значения этого типа достаточно 1 бита, но в реальности память такими порциями не выделяется, поэтому переменные этого типа могут быть по-разному упакованы виртуальной машиной	false либо true	false

Преобразование (приведение) типов - это процесс, при котором меняется тип выражения или переменных в выражении, может производиться автоматически либо указываться явно (тип данных указывается в круглых скобках перед литералом, переменной в выражении или перед выражением, но в таком случае выражение берется в круглые скобки). Преобразование типов отграничено для для некоторых типов данных или невозможно вообще.

На диаграмме ниже представлена возможность преобразования типов данных, где сплошные стрелки обозначают возможность преобразования без потери точности, а штрихованные указывают на возможную ее потерю.



Практические задания.

1. Создать программу, которая производить преобразования, указанные на диаграмме и производит вывод значений на экран.
2. Создать программу, которая меняет значения двух целочисленных переменных местами, не используя третью переменную.
3. Создать программу, которая реализует функцию деления по модулю (деление с остатком, который является результатом и принимает значения 1 или 0).
4. Создать программу, которая проверяет варианты переполнения граничных значений для всех типов данных.