UNIVERSITY OF TWENTE.

PROGRAMMING: VALUES AND VARIABLES

FAIZAN AHMED

MODULE 2: SOFTWARE SYSTEMS

VARIABLE

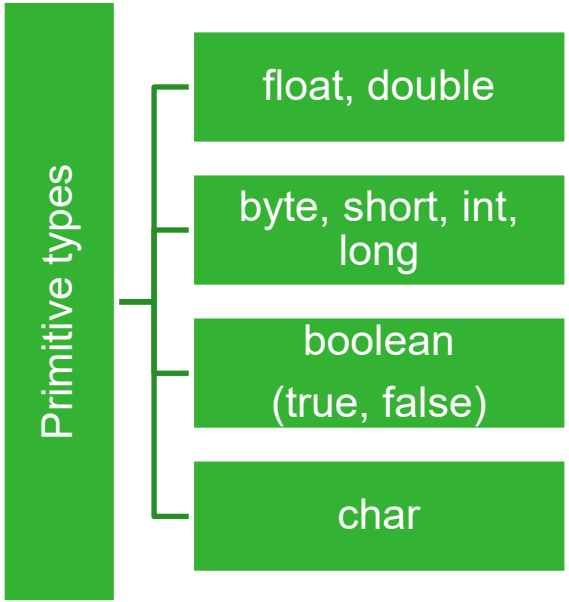
- Variable is a name for storage space that is used to store data

```
int count=10;
```

type name value

PRIMITIVE TYPES

- A variable in Java is designed to hold only one particular type of data;
  - Java is a strongly typed
- Primitive type
  - 8 primitive types



PRIMITIVE TYPE: CASTING

```
int anInteger;
double aDouble =
3.2;
anInteger=aDouble;
```

Error

```
int anInteger;
short aShort;
 anInteger=aShort;
```

Not an Error

PRIMITIVE TYPES: TYPE CASTING

Primitive type	size
double	64 bit (+/- 1.7 x 10308 with 15 significant digits)
float	32 bit (+/- 3.4 x 1038 with 7 significant digits)
long	64bit ($[-9 \times 10^{18}, 9 \times 10^{18}]$
int	32 bit ($[-2147483648, 2147483647]$)
short	16 bit ($[-32768,32767]$
byte	8 bit (-128,127)

Bottom to up conversion can be done automatically

```
int anInteger;
short aShort;
 anInteger=aShort;
```

PRIMITIVE TYPES: TYPE CASTING

- to force a conversion that wouldn't be done automatically a type cast is used
- A type cast is indicated by putting a type name, in parentheses, in front of the value you want to convert.

```
int anInteger;
double aDouble = 3.2;
anInteger = (int) aDouble
```

Type-casting change the value of a number

- type casts from any numeric type to any other numeric type

PRIMITIVE TYPES: TYPE CASTING

- char to int - return ASCII value

```
int a=(int) '-'; // the ASCII value 45
```

- int to char- returns the character

```
(char) 100; // it is letter d
```

CONSTANTS

‘VARIABLE’ WITH VALUES THAT CANNOT BE CHANGED

```
public static final int ROOK = 0;
public static final int KNIGHT = 1;
public static final int BISHOP = 2;
```

- Declared as final (static)
- Purpose: Understandability and maintainability
- Use constants if possible!

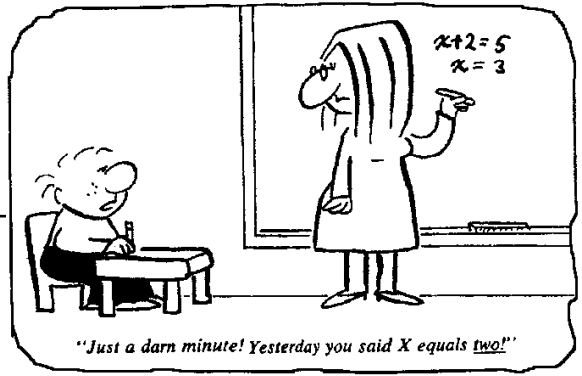
```
public static final int M2S = 60; // minutes to seconds
public static final int H2M = 60; // hours to minutes
...
int duration = 3215;
int sec = duration % M2S;
int min = duration / M2S;
int hr = min / H2M;
min = min - H2M*hr;
```

NAMING CONVENTIONS: LIKE IT OR LUMP IT!

NAMES OF DIFFERENT CODE ELEMENTS SHAPED DIFFERENTLY

- Recognition and thus readability → apply and get used to it!
- Choose meaningful names
  - Preferably whole words or traceable abbreviations
- Class names always start Uppercase
- variable names always start lowercase
- Constant names are all caps (only UPPERCASE)
- Names areCamelCase and_not_underscore
  - Except CONSTANT_NAMES (where CAMELCASE doesn't work)

VARIABLES



- We want a way to refer to an unknown value
- Variable is a name for storage space of a value
- Variables are typed → only accept values of certain type
- Variables must be declared before being used
- Variables can change their value through assignment
- At runtime, variables are stored in memory

In Java

```
int count = 10;
```

declaration assignment

In memory

count | 10

reference value