



## Para saber más: Type Casting

Como se ve en los videos, cuando intentamos poner un valor entero en una variable de tipo double, Java no muestra un error. Sin embargo, cuando intentamos poner un doble en una variable del tipo entero, tenemos un error de compilación.

Esta propiedad se produce porque Java convierte implícitamente de un tipo más pequeño a tipos "más grandes". De entero a double, por ejemplo.

Lo contrario no es cierto porque hay pérdida de datos cuando se realiza la conversión. Resultando en un "type mismatch" que muestra que esta instrucción es de tipos incompatibles.

Para realizar una conversión donde puede haber pérdida de información, es necesario hacer un type casting. Vea las instrucciones a continuación.

```
int edad = (int) 30.0;
```

[COPIA EL CÓDIGO](#)

En el caso anterior, es explícito que se realizará la conversión de double a entero. Vea cómo funciona el cast implícito y explícito en la tabla a continuación.

DE/PARA	byte	short	char	int	long	float	double
byte	----	<i>Impl.</i>	(char)	<i>Impl.</i>	<i>Impl.</i>	<i>Impl.</i>	<i>Impl.</i>
short	(byte)	----	(char)	<i>Impl.</i>	<i>Impl.</i>	<i>Impl.</i>	<i>Impl.</i>
char	(byte)	(short)	----	<i>Impl.</i>	<i>Impl.</i>	<i>Impl.</i>	<i>Impl.</i>
int	(byte)	(short)	(char)	----	<i>Impl.</i>	<i>Impl.</i>	<i>Impl.</i>
long	(byte)	(short)	(char)	(int)	----	<i>Impl.</i>	<i>Impl.</i>
float	(byte)	(short)	(char)	(int)	(long)	----	<i>Impl.</i>
double	(byte)	(short)	(char)	(int)	(long)	(float)	----

Para comparar cada tipo primitivo más claramente, la siguiente tabla muestra el tamaño de cada uno.

TIPO	TAMAÑO
boolean	1 bit
byte	1 byte
short	2 bytes
char	2 bytes
int	4 bytes
float	4 bytes
long	8 bytes
double	8 bytes