OCA – 3

* Creazione e manipolazione di stringhe

Una stringa è essenzialmente una sequenza di caratteri come nell’esempio che segue:

String foo = ”thinghs”;

String è un tipo riferimento, ovvero significa che la variabile dichiarata nell’esempio sopra con nome foo, non contiene direttamante un valore ma contiene l’indirizzo in memoria (cioè il riferimeto) alla stringa “things”.

“things”

Ox12b4

“things”

foo foo

Ox12b4

Memoria (heap)

Sappiamo che tipi riferimento servono per contenere degli oggetti che possono essere creati mediante la parola chiave del linguaggio che è ‘new’.

Pero’ noi nell’esempio precedente non abbiamo usato l’operatore new per creare un oggetto di tipo String.

Abbiamo quindi fatto un errore ?

No nessun errore. In effetti in java abbiamo 2 modi per creare un oggetto di tipo String:

1. scrivendo direttamente la stringa tra doppie virgolette come abbiamo visto prima

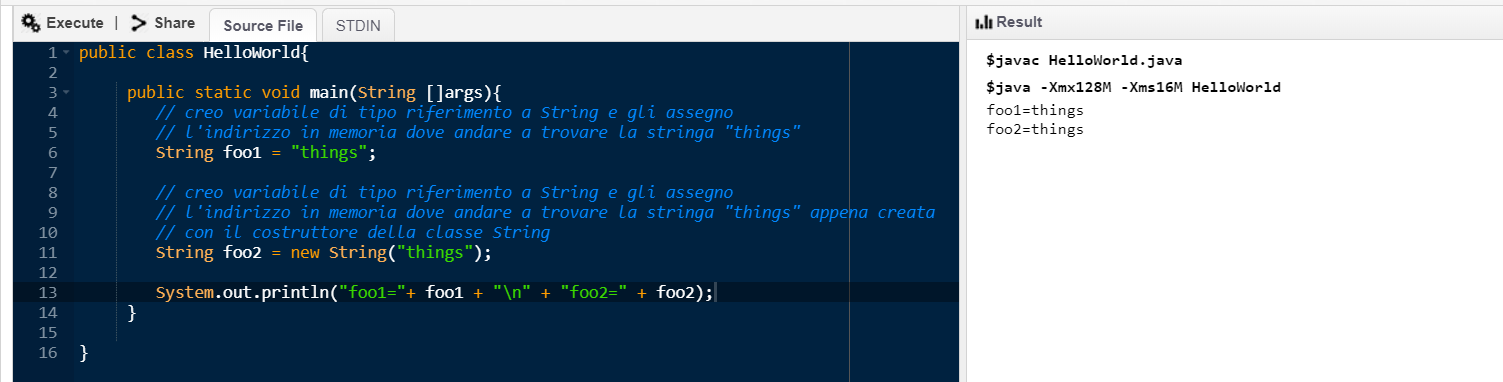
esempio : String foo1 = ”things”;

oppure

1. mediante l’operatore new, ovvero invocando il costruttore della classe String

esempio: String foo2 = new String(“things”);

Tutti e 2 le modalità creano una stringa “things” e assegnano il riferimento in memoria alla variabile battezzata foo proprio come nella figura mostrata sopra.



In verità le 2 modalità di creazione di una stringa nascondono una sottile differenza che vedremo nei prossimi paragrafi e che prende il nome di ‘String pool’.

* Concatenazione
* Immutabilità di una stringa

Una volta che è stato creato un oggetto di tipo String, non può più essere modificato.

* String pool

Dato che le string sono ovunque in java, esse possono occupare parecchia memoria.

In alcuni casi di applicazioni reali le stringhe possono occupare tra il 25 e il 40 percento di tutta la memoria della JVM. Per risolvere questo problema di spreco di memoria, java quello che fa è riusare le stringhe che si ripetono riutilizzando quindi le stringhe più comuni senza istanziarne delle nuove.

Lo ‘String pool’ che tradotto vuol dire piscina di stringhe, come vuol fare intendere la parola stessa, è una zona di memoria della JVM(quindi una zona dello heap, zona dove risiedono gli oggetti java) dove vengono salvate tutte queste stringhe.

Nota: Nello String pool vengono salvate tutti i letterali stringa come ad esempio “name”, “things”, ecc.

Mentre new String(“another thing”) non viene salvata nello String pol perché è un oggetto e non un letterale. Stessa cosa per myObject.toString(), questa non viene salvata nello String pool perché non è un letterale.

Quindi tornando indietro a quando abbiamo visto le 2 modalità per creare una stringa che erano :

1. scrivendo direttamente la stringa tra doppie virgolette come abbiamo visto prima

esempio : String foo1 = ”things”;

oppure

1. mediante l’operatore new, ovvero invocando il costruttore della classe String

esempio: String foo2 = new String(“another foo”);

La differenza nell’usare il primo o il secondo metodo è che con il primo metodo cioè 1), abbiamo un letterale stringa a destra dell’assegnamento e quindi la stringa andrà a far parte dello String pool,

mentre nel secondo caso cioè il 2) no in quanto a destra non abbiamo un letterale stringa.

Indirizzo Valore

“things”

Ox4c3

Ox12b4

foo1 Ox12d8

Ox12d8

foo2

Ox094

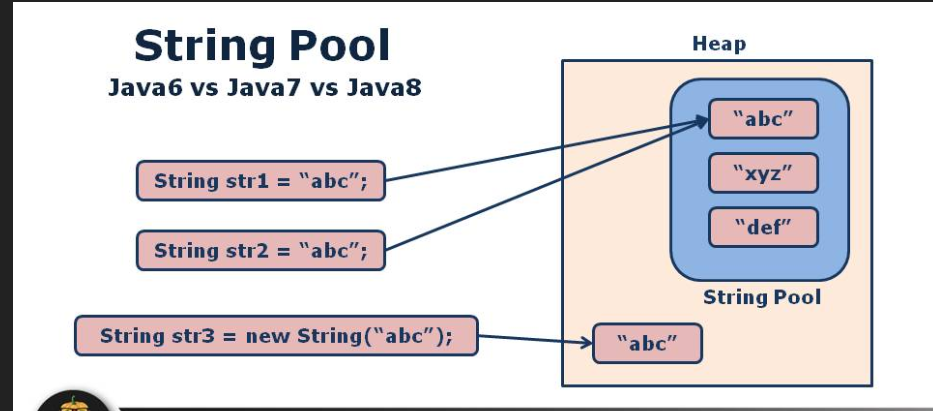
.

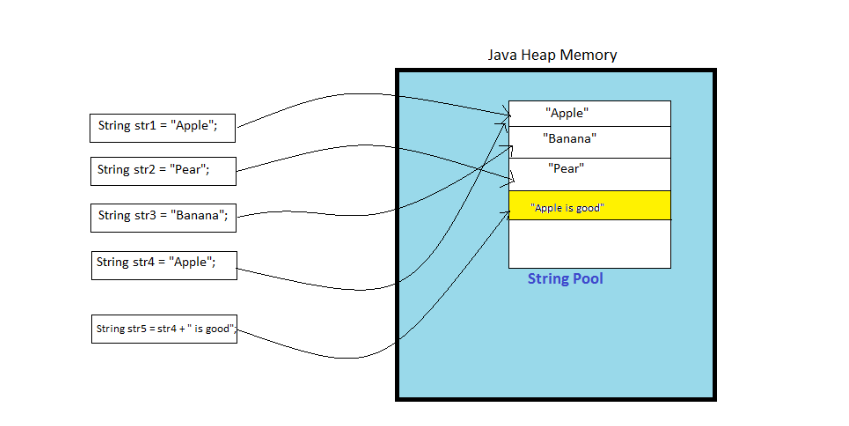
STRING POOL . .

.

Ox02a

Memoria (Heap)





* Metodi importanti della classe String