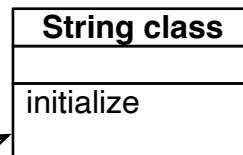
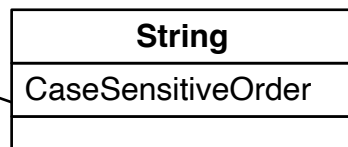


```
compare: aString caseSensitive: aBool  
  | map |  
  map := aBool ifTrue: [ CaseSensitiveOrder ]  
  ifFalse: [ CaseInsensitiveOrder ].  
  ^self compare: self with: aString collated: map
```



instanceOf

String **class** >> initialize

```
self initializeTypeTable.
```

```
AsciiOrder := self newAsciiOrder.
```

```
CaseInsensitiveOrder := self newCaseInsensitiveOrder.
```

```
CaseSensitiveOrder := self newCaseSensitiveOrder.
```

```
LowercasingTable := self newLowercasingTable.
```

Symbol **class** >> selectorsContaining: aString

...

```
self allSymbolTablesDo: [:each |
```

```
  each size >= size ifTrue:
```

```
    [(each findSubstring: aString in: each startingAt: 1
```

```
      matchTable: CaseInsensitiveOrder) > 0
```

```
      ifTrue: [selectorList add: each]]].
```

...