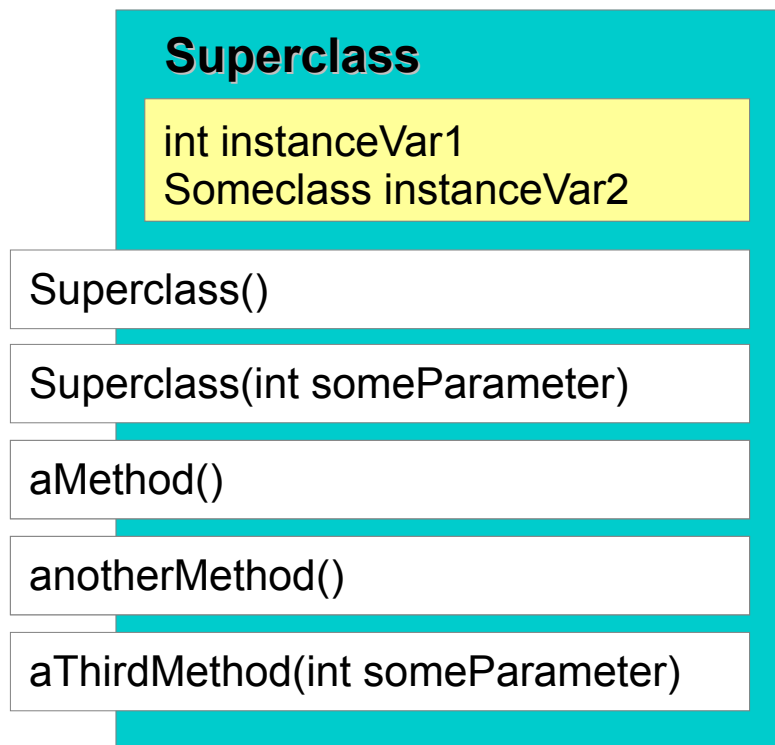
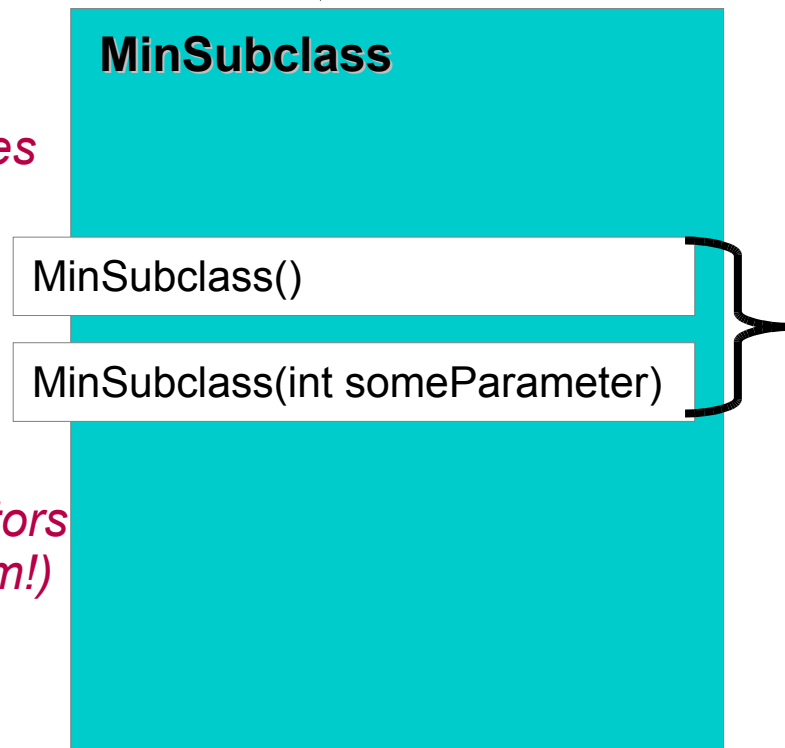




A subclass definition specifies new state & methods, and inherits everything else.

Must always specify constructors (can't inherit them!)

This minimal subclass only specifies constructors; inherits everything else.



constructors

Superclass

```
int instanceVar1  
Someclass instanceVar2
```

```
Superclass()
```

```
Superclass(int someParameter)
```

```
aMethod()
```

```
anotherMethod()
```

```
aThirdMethod(int someParameter)
```



*Here we see the
inherited state
and methods...*

*... as well as
the new
constructors.*

MinSubclass

```
int instanceVar1  
Someclass instanceVar2
```

```
MinSubclass()
```

```
MinSubclass(int someParameter)
```

```
aMethod()
```

```
anotherMethod()
```

```
aThirdMethod(int someParameter)
```

inherited
state

constructors

inherited
methods

Superclass

int instanceVar1
Someclass instanceVar2

Superclass()

Superclass(int someParameter)

aMethod()

anotherMethod()

aThirdMethod(int someParameter)

The new subclass has additional state and methods.

MinSubclass

int instanceVar1
Someclass instanceVar2

inherited state

constructors

MinSubclass()

MinSubclass(int someParameter)

aMethod()

anotherMethod()

aThirdMethod(int someParameter)

inherited methods

RicherSubclass

int instanceVar1
Someclass instanceVar2
int newInstanceVar

new state

RicherSubclass()

RicherSubclass(int someParam)

aMethod()

anotherMethod()

aThirdMethod(int somePa

new methods

aFourthMethod()

yetAnotherMethod(int aParameter)

Superclass

```
int instanceVar1  
Someclass instanceVar2
```

```
Superclass()
```

```
Superclass(int someParameter)
```

```
aMethod()
```

```
anotherMethod()
```

```
aThirdMethod(int someParameter)
```

MinSubclass

constructors

```
MinSubclass()
```

```
MinSubclass(int someParameter)
```

RicherSubclass

new
state

```
int newInstanceVar
```

constructors

```
RicherSubclass()
```

```
RicherSubclass(int somePa
```

new
methods

```
aFourthMethod()
```

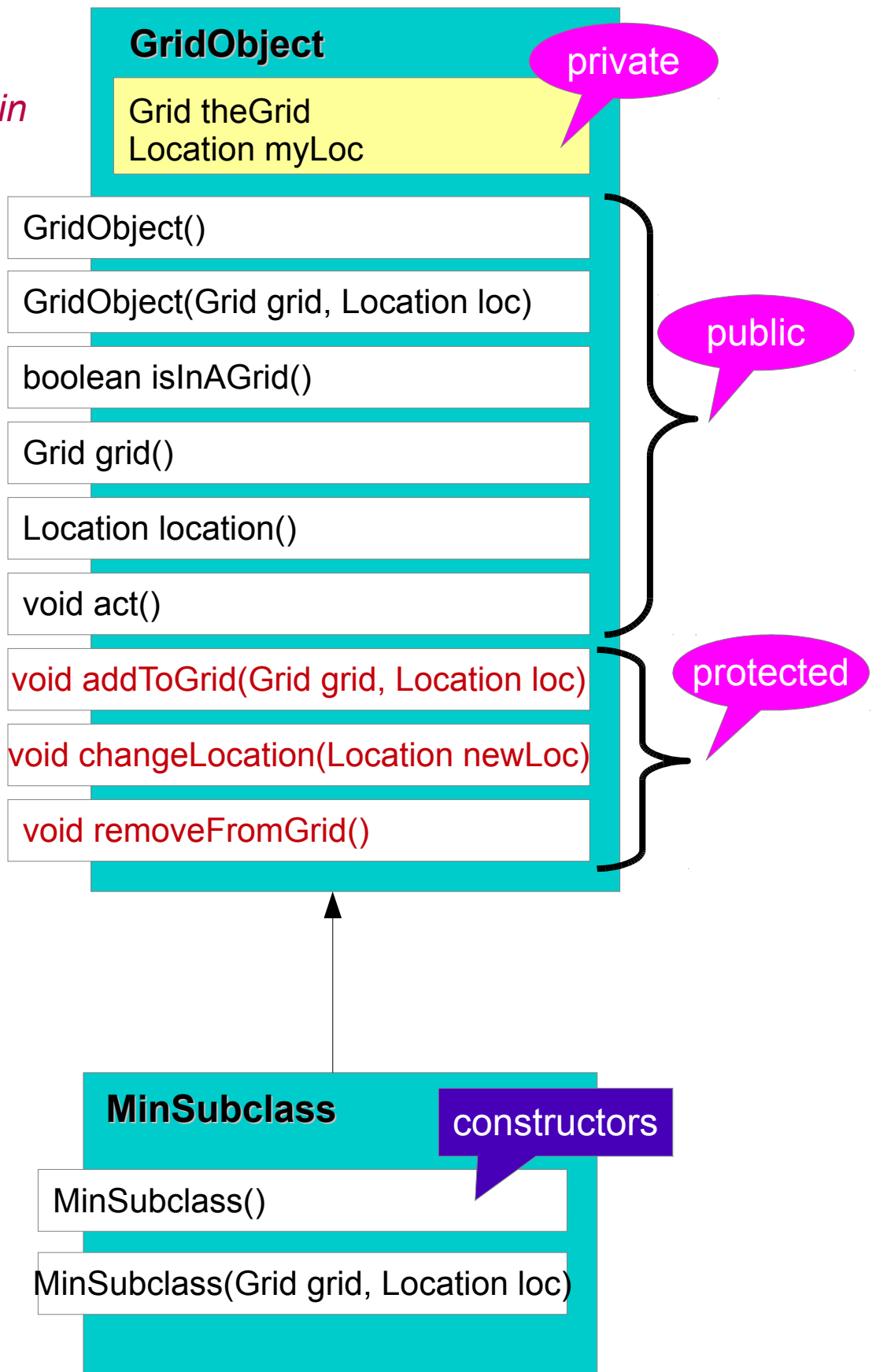
```
yetAnotherMethod(int aParameter)
```

*We usually don't show the
inherited state & methods,
just the new ones.*

Private: only accessible within this class.

Public: accessible to code in any class.

Protected: accessible to code in this or any subclass.



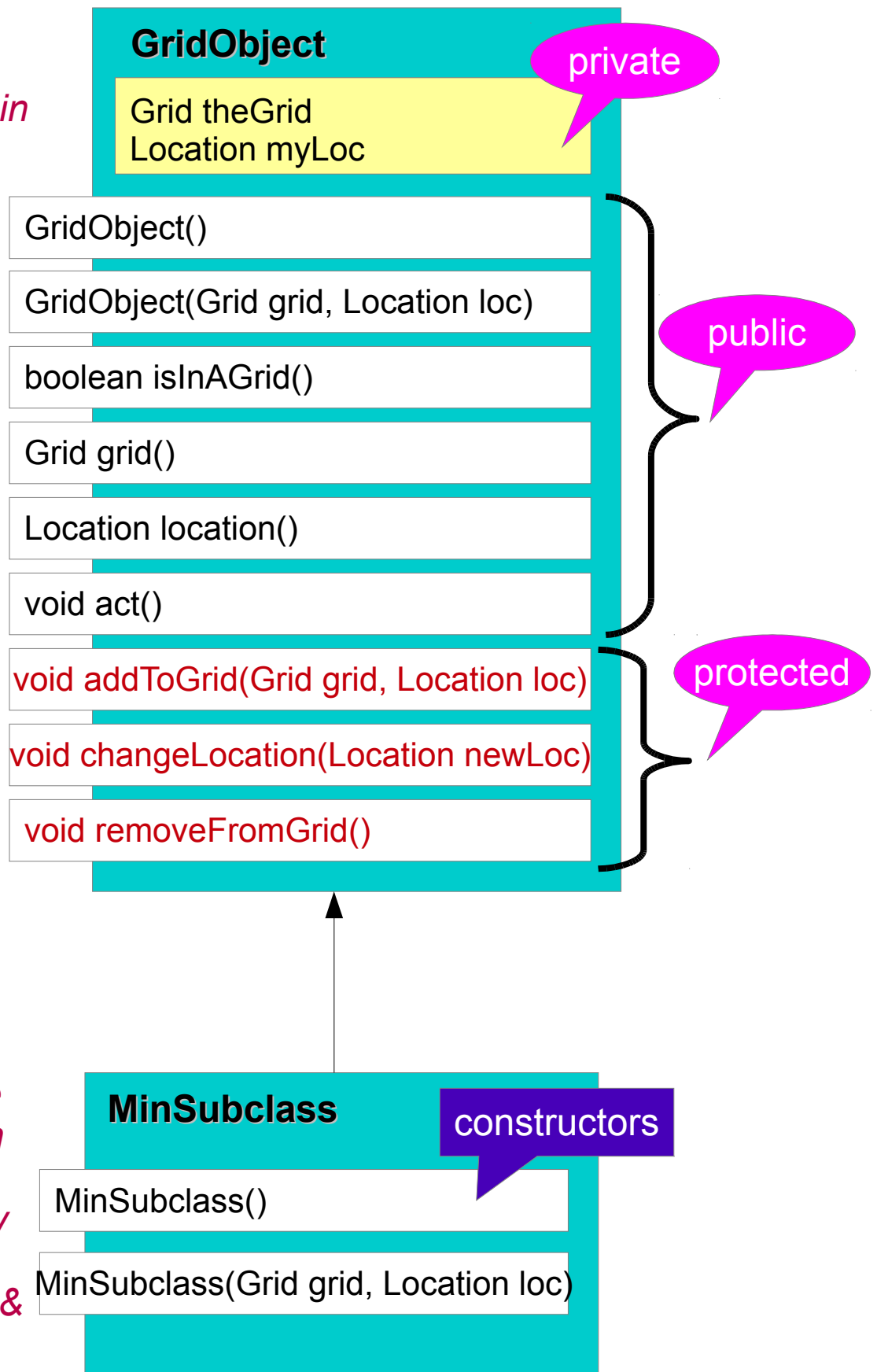
Private: only accessible within this class.

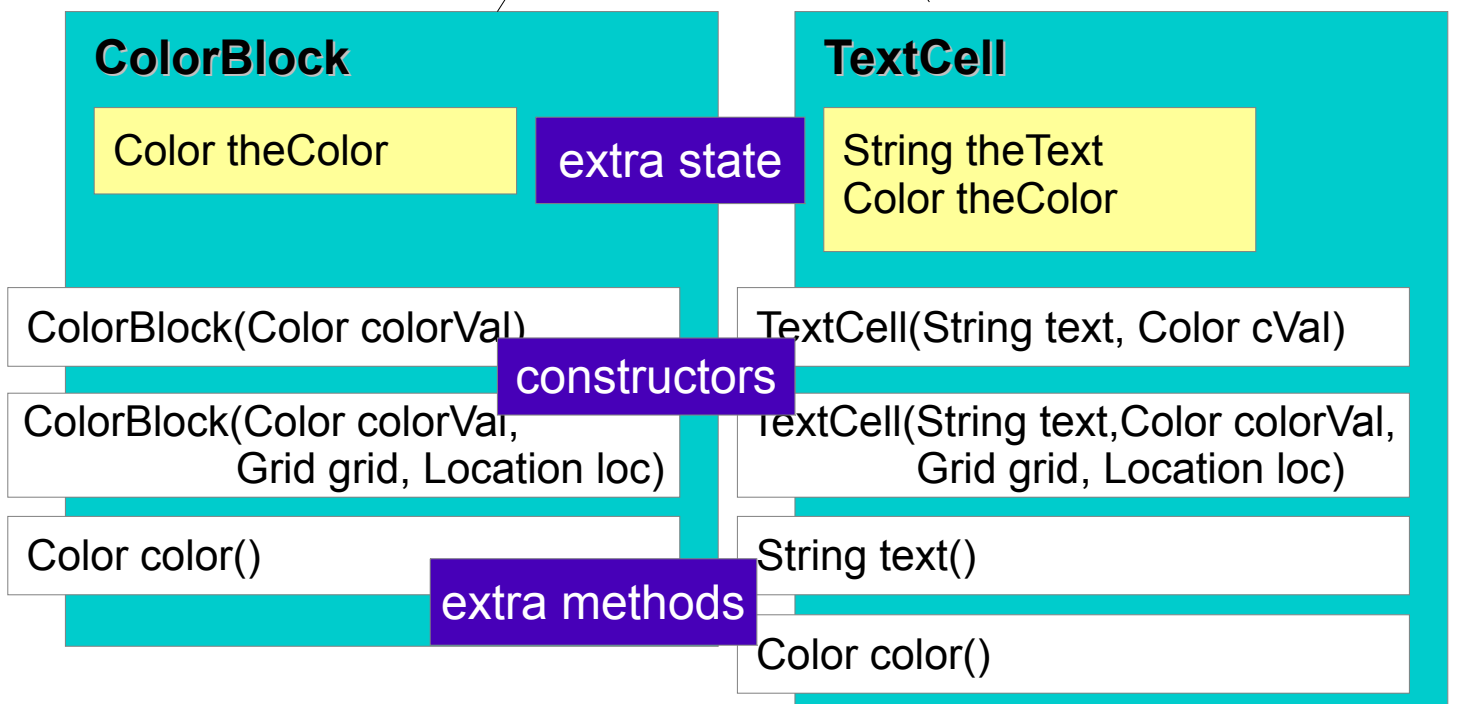
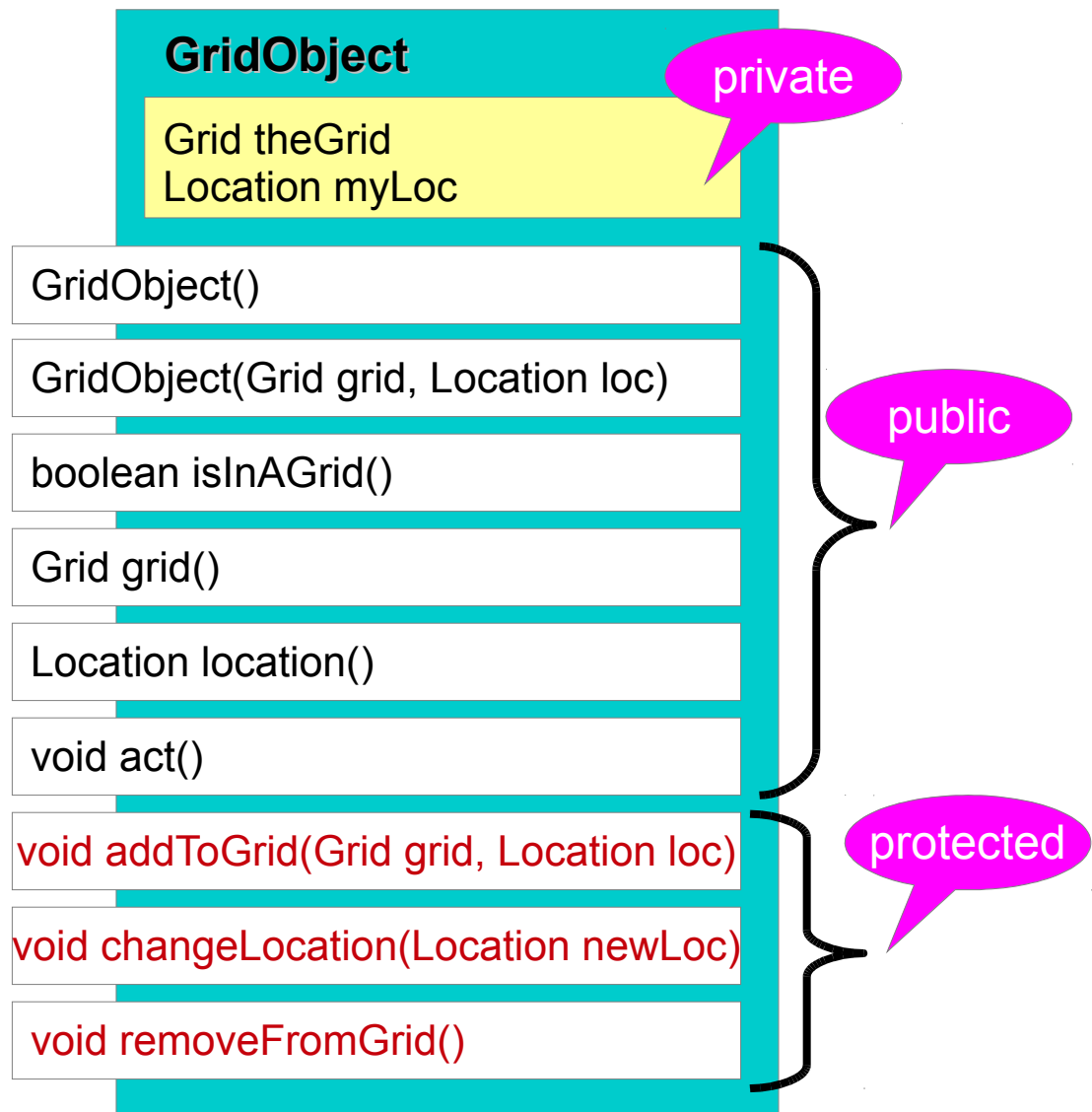
Public: accessible to code in any class.

Protected: accessible to code in this or any subclass.

Inherits all state & methods from superclass, but can only directly use public and protected state & methods.

Can access private inherited state only through methods in the superclass.





GridObject

Grid theGrid
Location myLoc

GridObject()

GridObject(Grid grid, Location loc)

boolean isInAGrid()

Grid grid()

Location location()

void act()

void addToGrid(Grid grid, Location loc)

void changeLocation(Location newLoc)

void removeFromGrid()

ColorBlock

Color theColor

ColorBlock(Color colorVal)

ColorBlock(Color colorVal,
Grid grid, Location loc)

Color color()

TextCell

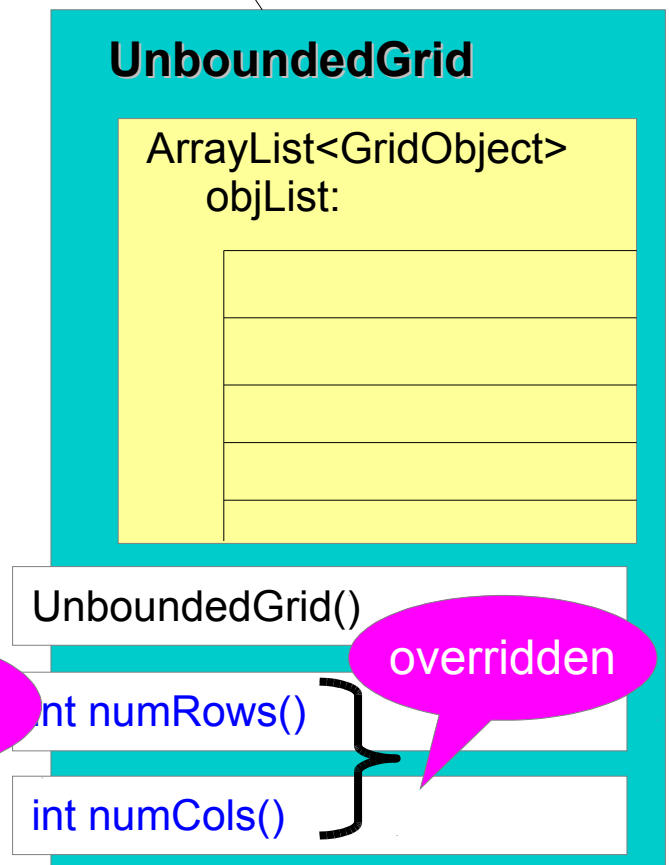
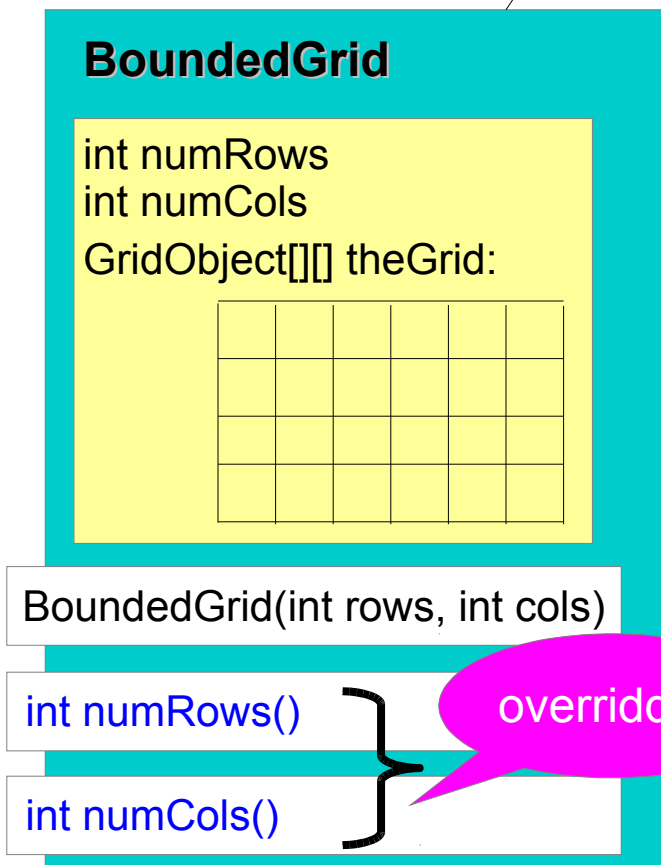
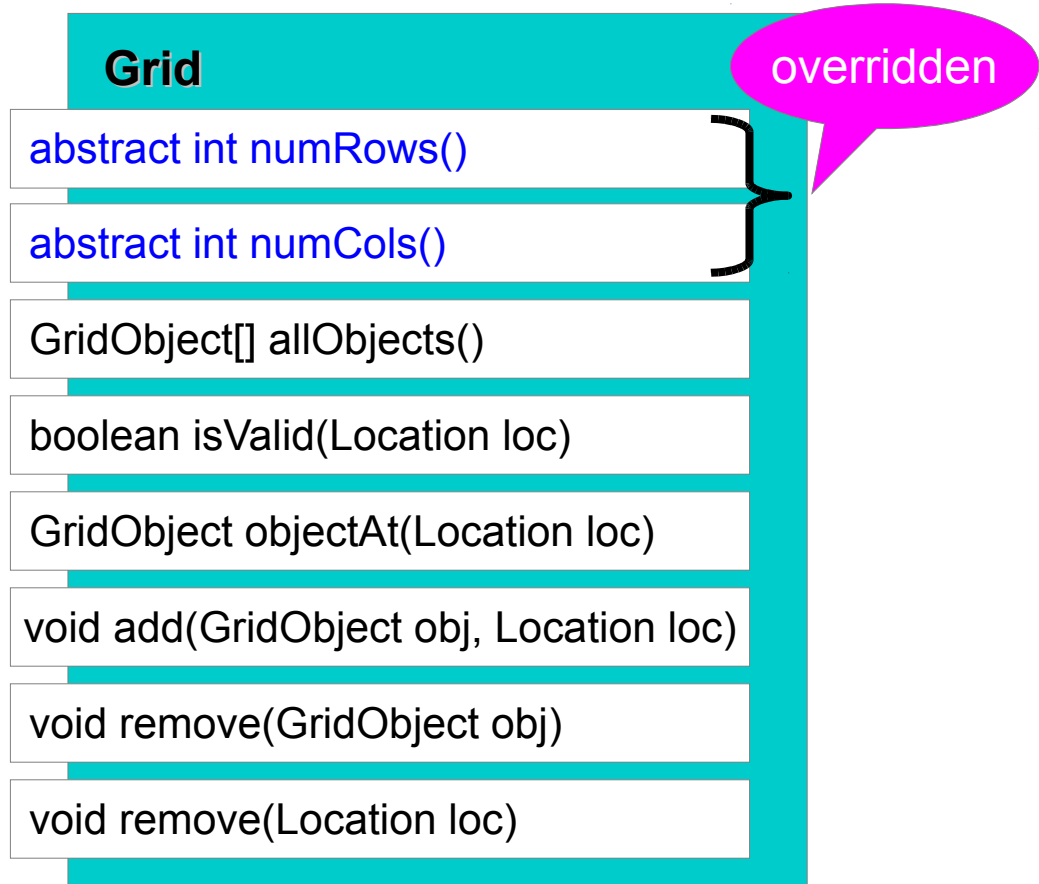
String theText
Color theColor

TextCell(String text, Color cVal)

TextCell(String text,Color colorVal,
Grid grid, Location loc)

String text()

Color color()



Grid

abstract int numRows()

abstract int numCols()

GridObject[] allObjects()

boolean isValid(Location loc)

GridObject objectAt(Location loc)

void add(GridObject obj, Location loc)

void remove(GridObject obj)

void remove(Location loc)

BoundedGrid

int numRows

int numCols

GridObject[][] theGrid:

BoundedGrid(int rows, int cols)

int numRows()

int numCols()

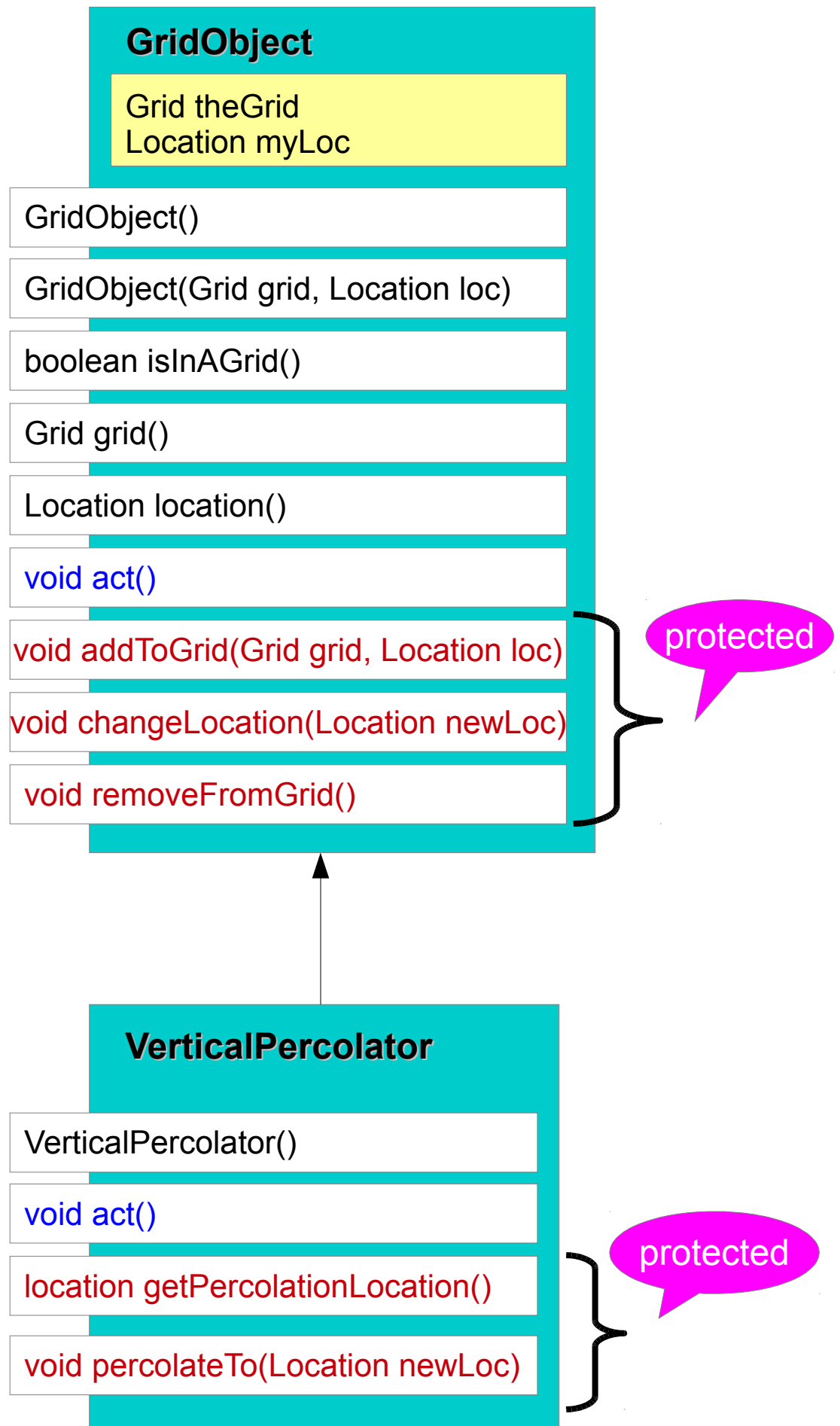
UnboundedGrid

ArrayList<GridObject>
objList:

UnboundedGrid()

int numRows()

int numCols()



GridObject

Grid theGrid
Location myLoc

GridObject()

GridObject(Grid grid, Location loc)

Grid grid()

Location location()

void act()

⋮

AbstractPercolator

AbstractPercolator()

void act()

ArrayList<Location>
getPercolationLocations()

void percolateTo(Location
newLoc)

GridObject duplicate()

VerticalPercolator

VerticalPercolator()

GridObject duplicate()

AllDirectionPercolator

AllDirectionPercolator()

ArrayList<Location>
getPercolationLocations()

GridObject duplicate()

GravitationalPercolator

GravitationalPercolator()

ArrayList<Location>
getPercolationLocations()

GridObject duplicate()

