

SingleBandShortPixelImage

coverage : GridCoverage

```
<<create>> SingleBandShortPixelImage(coverage : GridCoverage,colorModel : ColorModel) {sequential}  
getData() : Raster {sequential}  
getTile(x : int,y : int) : Raster {sequential}
```

StretchPixelColorModel

colors : Color[]
factor : float
min : int
max : int
diff : int
nodataColor : Color
nodataValues : int[]

```
<<create>> StretchPixelColorModel(min : int,max : int,nodataValues : int[],nodataColor : Color) {sequential}  
setMinMaxValue(min : int,max : int) : void {sequential}  
getMinValue() : int {sequential}  
getMaxValue() : int {sequential}  
getAlpha(pixel : int) : int {sequential}  
getAlpha(inData : Object) : int {sequential}  
getBlue(pixel : int) : int {sequential}  
getBlue(inData : Object) : int {sequential}  
getGreen(pixel : int) : int {sequential}  
getGreen(inData : Object) : int {sequential}  
getRed(pixel : int) : int {sequential}  
getRed(inData : Object) : int {sequential}  
isCompatibleSampleModel(sm : SampleModel) : boolean {sequential}  
isCompatibleRaster(raster : Raster) : boolean {sequential}  
getColorValue(pixel : int) : int {sequential}  
getColorValueExp(pixel : int) : int {sequential}  
getColorValuePow(pixel : int) : int {sequential}  
getColorValueLog(pixel : int) : int {sequential}  
getColorValueSimple(pixel : int) : int {sequential}  
getColorValue(inData : Object) : int {sequential}  
isNoData(pixel : int) : boolean {sequential}
```