

-
-
+
+
+
+
+
+
+



<<Shape>>

Line

start: QPoint

end: QPoint

```
Line(*device:QPaintDevice=nullptr,TempID:in
      TempPen:QPen=NoPen,TempBrush:QBrush=No
      TempPoint1:QPoint=(0,0),TempPoint2:QPo
      0))
```

```
draw(in *device:QPaintDevice): void
```

```
move(in *source:Shape): void
```

```
getStart(): QPoint
```

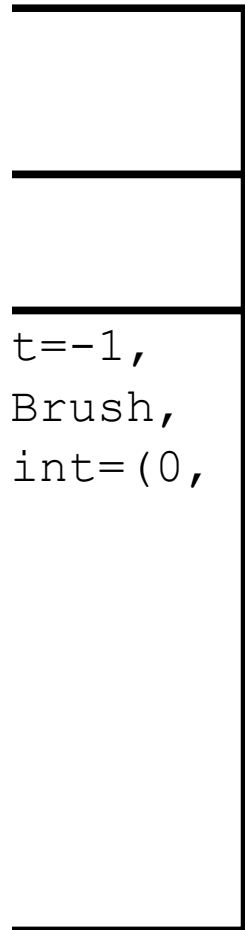
```
getEnd(): QPoint
```

```
calcPerimeter(): double
```

```
calcArea(): double
```

<<Shape>>

Defining



t=-1,
Brush,
int=(0,



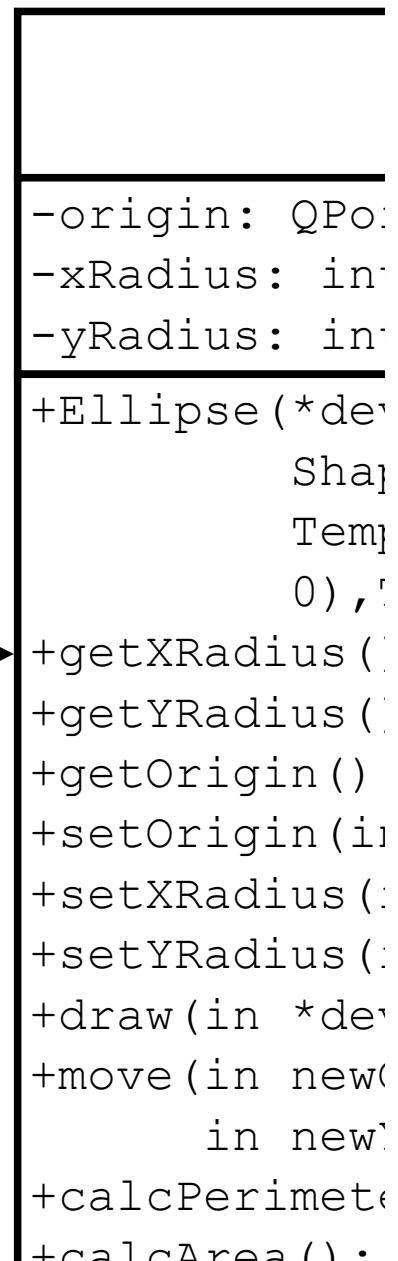
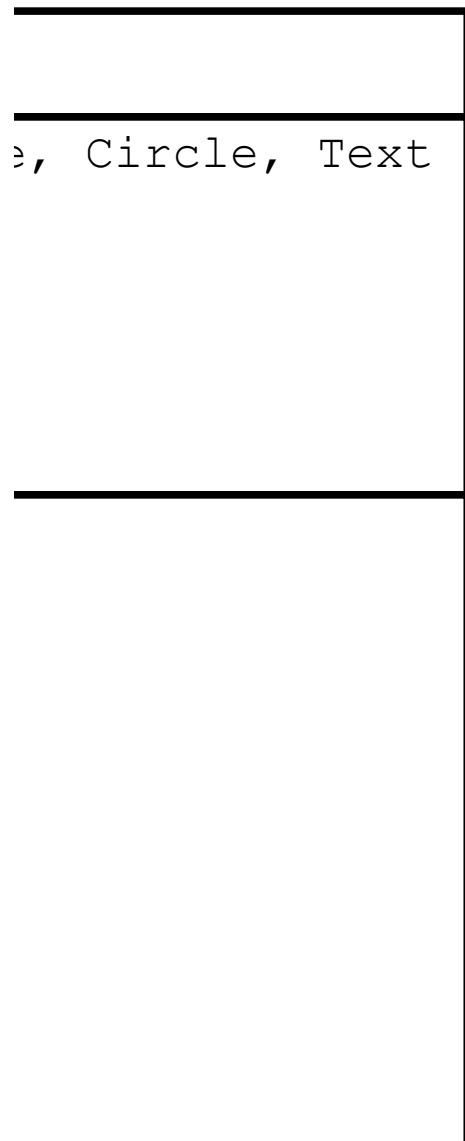
+shape: enum = Line, Pc
-painter: QPainter
-ID: int
-objShape: shape
-Pen: QPen
-Brush: QBrush

+Shape(*device:QPaintDe
TempShape:shape=
TempBrush:QBrush
+setID(in InID:int): vc
+setShape(in InShape:sh
+setPen(in &InPen:QPen)
+setBrush(in &InBrush:C
+getID(): int
+getShape(): shape
+&getPen(): QPen
+&getBrush(): QBrush

Shape

polyline, Polygon, Rectangle, Square, Ellipse

```
evice=nullptr, InID:int=-1,  
=Line, TempPen:QPen=NoPen,  
1=NoBrush)  
oid  
ape): void  
 : void  
Brush): void
```



<<Shape>>

Ellipse

i: int

t

t

vice: QPaintDevice, TempID: int = -1,
pe: shape = Ellipse, TempPen: QPen = NoPen,
pBrush: QBrush = NoBrush, TempOrigin: QPoint = (0,
TempXRadius: int = 0, TempYRadius: int = 0)

): int

): int

: QPoint

n NewOrigin: QPoint): void

in NewXRadius: int): void

in newYRadius: int): void

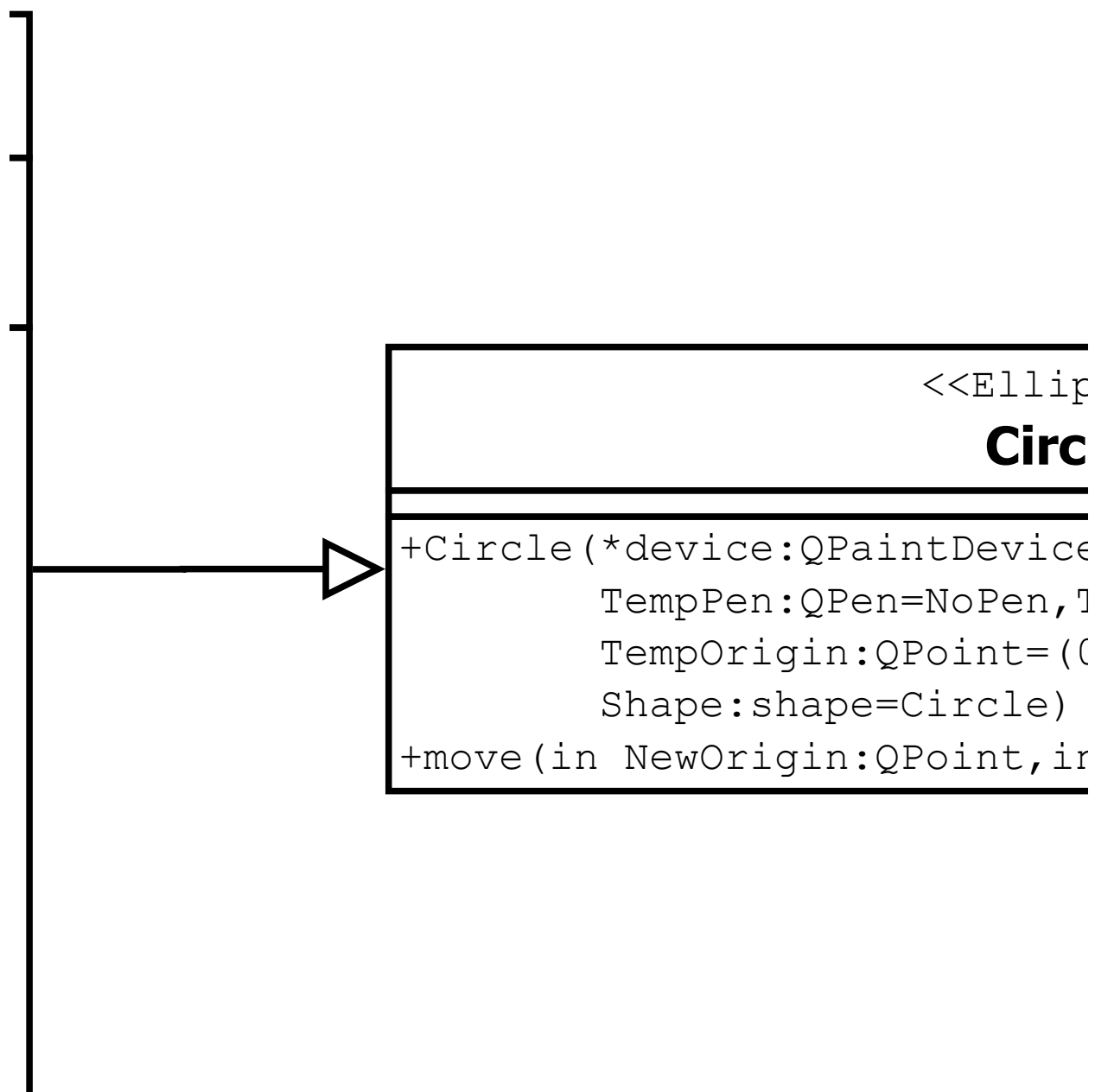
vice: QPaintDevice): void

Origin: QPoint, in newXRadius: int,

YRadius: int): void

er(): double

double



-Pc

+Pc

+dr

+ge

+mc

+cā

+cā

Polyline

```
pointVector: vector<QPoint>
Polyline(*device:QPaintDevice=nullptr,TempID:
        TempPen:QPen=NoPen,TempBrush:QBrush=
        &TempVector:vector<QPoint>=vector<QPoint>(),
draw(in *device:QPaintDevice): void
getPoints(): vector<QPoint>
move(in *source:Shape): void
calcPerimeter(): double
calcArea(): double
```

-Po

+Po

+dr

+mo

+ca

+ca

```

: int=-1,
=NoBrush,
?oint>())

```

```

+<<virtual>> getStart()
+<<virtual>> getEnd():
+<<virtual>> getPoints()
+<<virtual>> draw(in *device:QPaintDevice)
+<<virtual>> move(in &source:vector<QPoint>)
+<<virtual>> calcPerimeter(): double
+<<virtual>> calcArea(): double
+<<virtual>> ~Shape()
#&getPainter(): QPainter

```

<<Shape>>

Polygon

```

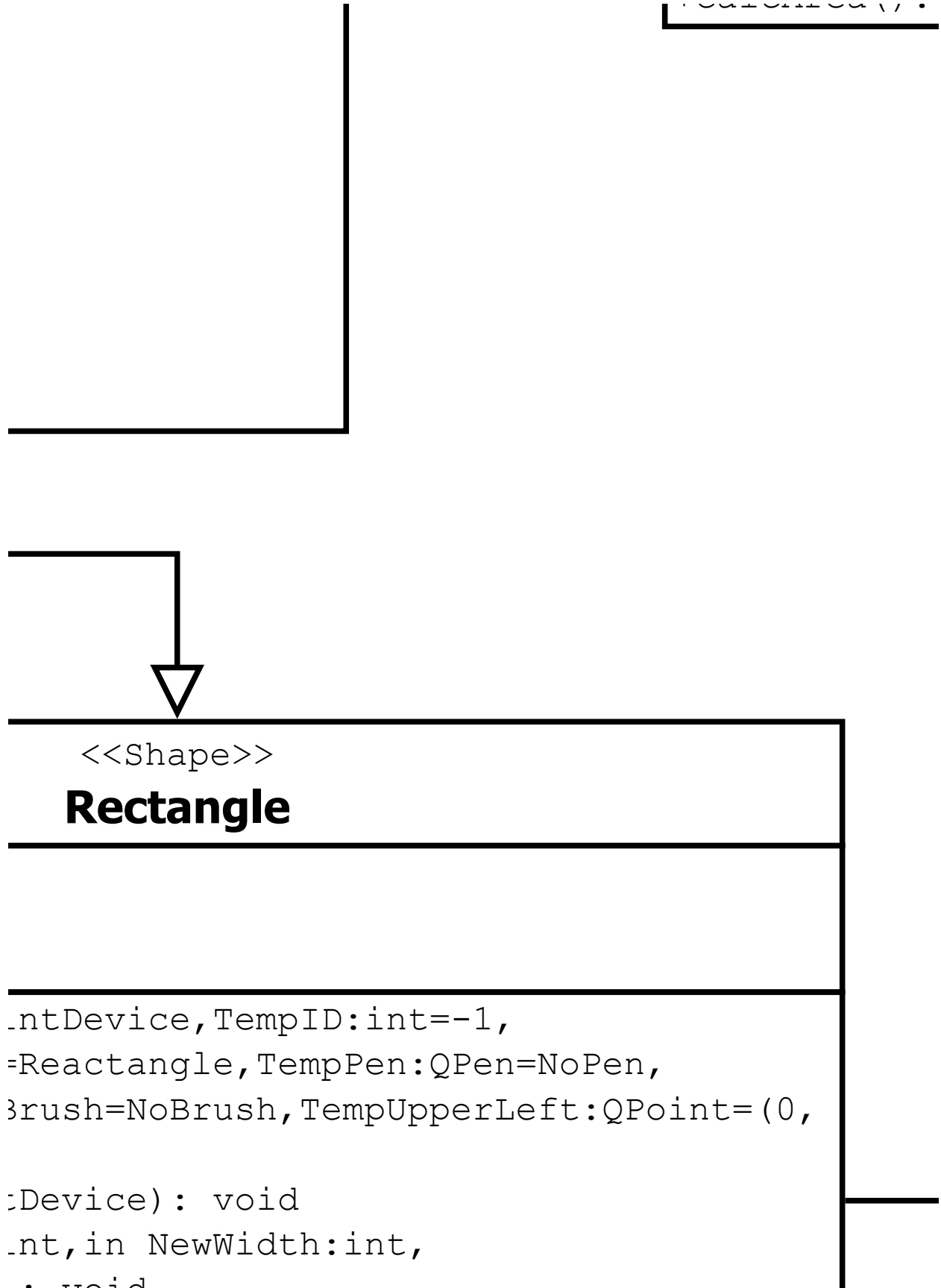
intVector: vector<QPoint>

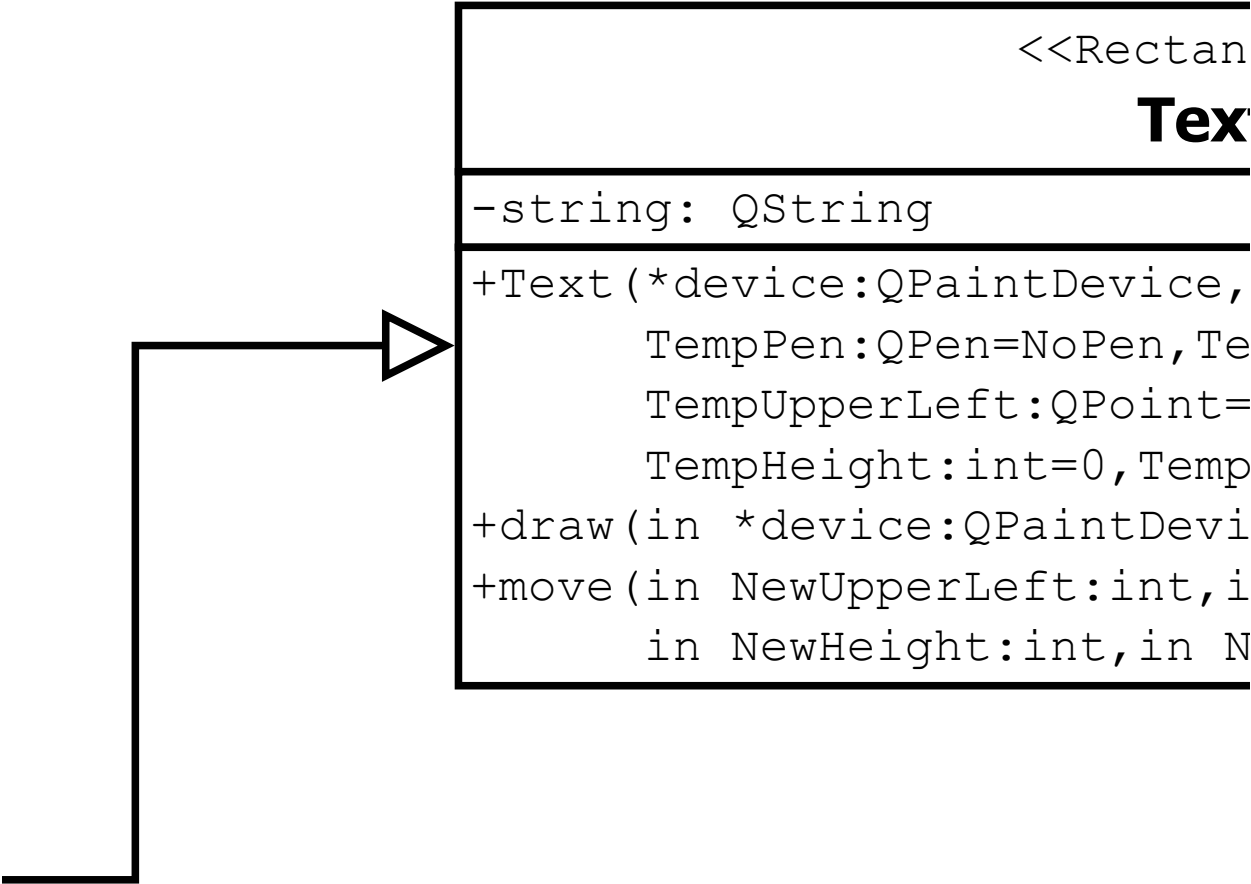
```

```

Polygon(*device:QPaintDevice,TempID:int=-1,
        TempPen:QPen=NoPen,TempBrush:QBrush=NoBrush,
        &TempVector:vector<QPoint>=vector<QPoint>())
draw(in *device:QPaintDevice): void
move(in &source:vector<QPoint>): void
calcPerimeter(): double
calcArea(): double

```



」

```
gle>>
```

```
t
```

```
TempID:int=-1,  
mpBrush:QBrush=NoBrush,  
(0,0),TempWidth:int=0,  
String:QString="")  
ce): void  
n NewWidth:int,  
ewString:QString): void
```



<<QWidget>>

canvas

1

vector

#size v. int

```

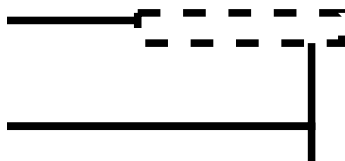
        in NewHeight.toInt(),
+calcPerimeter(): double
+calcArea(): double
+setUpperLeft(in NewUpperLeft: QPoint): void
+setWidth(in NewWidth: int): void
+setHeight(in NewHeight: int): void
+getUpperLeft(): QPoint
+getWidth(): int
+getHeight(): int

```

```

+Square(*device, TempPe, TempUp, Shape: ShapeType): void
+move(in NewPosition: QPoint): void

```



```

<<QWidget>>
RenderArea

```

```

    . void
    .e

    perLeft:QPoint): void
    nt): void
    :int): void
    ;

```



<<Rectangle>>

Square

```

ce:QPaintDevice,TempID:int=-1,
en:QPen=NoPen,TempBrush:QBrush=NoBrush,
perLeft:QPoint=(0,0),side:int=0,
shape=Square)
oint:QPoint,in NewSide:int): void

```

<<QWidget

Windo

t>>

...

+sh
+*c
+cā
+se
+ge
+ge
+pā

+*
+M
+s
+p
+r
+~
-o
-o

```
shapes: vector<Shape::Shape*>
```

```
canvasArea: canvas
```

```
Canvas (parent:QWidget*=0)
```

```
setShape (source:vector<Shape::Shape*>): void
```

```
getShapeCount(): int
```

```
getShape(int): Shape::Shape*
```

```
paintEvent (*e:QPaintEvent): void
```

<<QMainWindow>>

MainWindow

```
ui: MainWindow
```

```
MainWindow (*parent:QWidget=0)
```

```
setShape (vector<Shape::Shape*>): void
```

```
paintEvent (*e:QPaintEvent): void
```

```
refreshIds(): void
```

```
MainWindow()
```

```
on_shapeIdModSpinBox_valueChanged(arg1:int):
```

```
on_lineModMenu_updateButton_clicked(): void
```

}

void

```
#size_v. inc
#elem: T*
#space: int

+vector()
+vector(s:int)
+vector(src:vector&)
+vector(source:vector&&)
+~vector()
+size(): int
+capacity(): int
+resize(newSize:int): void
+push_back(d:T): void
+reserve(newalloc:int): void
+begin(): iterator
+begin(): const_iterator
+end(): iterator
+end(): const_iterator
+insert(p:iterator, val:T&):
+erase(p:iterator): iterator
```


ld

: iterator
or

RenderArea

```
+shape: Shape
+pen: QPen
+brush: QBrush
+antialiased: bool
+transformed: bool
+pixmap: QPixmap

+RenderArea(*parent:QWidget
+minimumSizeHint(): QSize
+sizeHint(): QSize
+setShape(shape:Shape): void
+setPen(&pen:QPen): void
+setBrush(&brush:QBrush): void
+setAntialiased(antialiase
+setTransformed(tranformed
#paintEvent(*event:QPaintE
```

```
t=0)
```

```
id
```

```
void
```

```
d:bool): bool
```

```
:bool): void
```

```
vent ): void
```

WINDOW

```
+*renderArea: RenderA
```

```
+*shapeLabel: QLabel
```

```
+*penWidthLabel: QLab
```

```
+*penStyleLabel: QLab
```

```
+*penCapLabel: QLabel
```

```
+*penJoinLabel: QLabel
```

```
+*brushStyleLabel: QLa
```

```
+*otherOptionsLabel: Q
```

```
+*shapeComboBox: QCom
```

```
+*penWidthSpinBox: QS
```

```
+*penStyleComboBox: Q
```

```
+*penCapComboBox: QCC
```

```
+*penJoinComboBox: QC
```

```
+*brushStyleComboBox:
```

```
+*antialiasingCheckBo
```

```
+*transformationsChec
```

```
-shapedChanged(): vo
```

```
-penChanged(): void
```

```
-brushChanged(): voic
```

W

Area

el

el

-

abel

)Label

nboBox

spinBox

)ComboBox

boBox

boBox

: QComboBox

x: QCheckBox

ckBox: QCheckBox

id

l

