

Referència ràpida de Programació d'Esdeveniments (SIGNALS i SLOTS)

Concepte

- **Esdeveniments** (serveis per indicar que està passant alguna cosa!) → SIGNAL
Es defineixen **només** a l'arxiu de capçalera :

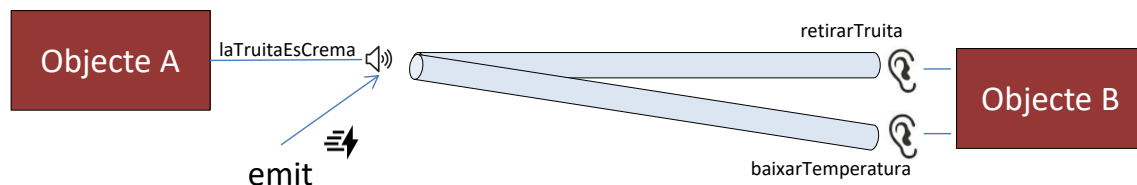
```
signals:  
    void laTruitaEsCrema(int temperatura);
```

- **Mètodes de gestió de l'esdeveniment** (Avisa'm si passa alguna cosa) → SLOTS
Es declaren a l'arxiu de capçalera, i s'implementen al cpp amb les accions que calgui emprendre.

```
public slots:  
    void baixarTemperatura(int value);  
    void retirarTruita(int value);
```

Posteriorment, cal connectar SIGNALS i SLOTS:

```
connect(this, SIGNAL(laTruitaEsCrema(int)), this, SLOT(retirarTruita(int)));  
connect(this, SIGNAL(laTruitaEsCrema(int)), this, SLOT(baixarTemperatura(int)));
```



Qt5 té una sintaxi nova per connectar SIGNALS i SLOTS:

```
connect(this, &ClasseOrigen::laTruitaEsCrema, this, &ClasseDesti::retirarTruita);  
connect(this, &ClasseOrigen::laTruitaEsCrema, this, &ClasseDesti::baixarTemperatura);
```

On heu de reemplaçar "ClasseOrigen" i "ClasseDesti" per les classes que contenen el signal i l'slot respectivament (poden ser la mateixa classe)

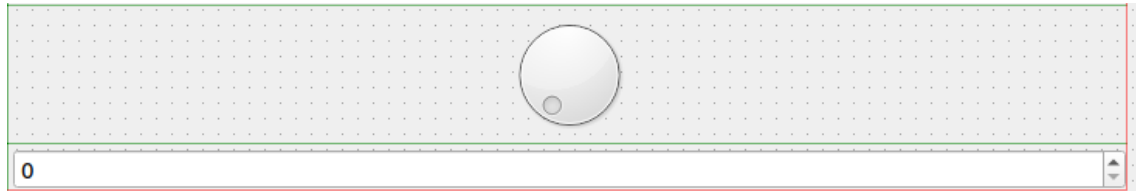
Per disparar un SIGNALS simplement "s'emet":

```
emit laTruitaEsCrema(value);
```

Connexió directa entre dos Widgets

Des de **QtCreator** podem connectar signals des de l'interfície gràfica.

Creeu un **QSpinBox** i un **QDial** al formulari:



Object	Class
Form	QWidget
gridLayout_2	QGridLayout
dial	QDial
spinBox	QSpinBox

Per connectar el SIGNAL de valor canviat del **QSpinBox** (`valueChanged(int)`) amb el SLOT d'assignar valor de **QDial** (`setValue(int)`), podem definir el **connect** des de la interfície gràfica de **QtCreator**:

Sender	Signal	Receiver	Slot
spinBox	valueChanged(int)	dial	setValue(int)

Alternativament també ho podem fer des de codi:

```
QObject::connect(ui.spinBox, SIGNAL(valueChanged(int)), ui.dial, SLOT(setValue(int)));
```

COMPTE: En aquest cas concret usar la nova sintaxi és un xic més complicat, donat que **QSpinBox::valueChanged** té dues sobrecàrregues, amb paràmetres *int* i amb *QString*. En aquest cas s'ha de fer una conversió a punter a funció prèviament per ajudar al compilador a decidir quina funció de les dues sobrecàrregues ha de triar.

```
void (QSpinBox::* valueChangedSignal)(int) = &QSpinBox::valueChanged;  
connect(ui.spinBox, valueChangedSignal, ui.dial, &QDial::setValue );
```