

Curso de aprendizaje de QtQuick



Modulo Cero



Objetivos

1 Yo

2 Qt

3 Javascript

Introducción

Sentencias

Comentarios

Variables

Operadores

Comparaciones

Condicionales

Funciones

Bucles: for, while, break

Objetos

Cadenas

Fechas

Vectores

Booleanos

Matemáticas

Temas

1 Yo

2 Qt

3 Javascript

Introducción

Sentencias

Comentarios

Variables

Operadores

Comparaciones

Condicionales

Funciones

Bucles: for, while, break

Objetos

Cadenas

Fechas

Vectores

Booleanos

Matemáticas

Objetivos

1 Yo

PAU GARCIA I QUILES

TELECOMMUNICATIONS ENGINEER (EE+CS)
SOFTWARE DEVELOPER

PGQUILES@ELPAUER.ORG
[HTTP://WWW.ELPAUER.ORG](http://www.elpauer.org)

```
001 @OPTIONS MAIN
002 Identification Division
003 Program-Id. elpauer.
004* String businesscard
005 Environment Division.
006 Configuration Section.
007 Source-Computer. SPAIN.
008 Object-Computer. WORLD.
009 Special-Names.
010   Currency-sign is EUR
011 Data Division.
012 Working-Storage Section.
013 Procedure-Division.
```



CD66 12E8 7145 C5B1 1EE6 3402 936A 835B 678B 9796

GPG fingerprint



Objetivos

1 Yo

QT rzzrdd **LINUX** kjir **C++** vksndwuo **MAEMO**
sz **BOOST** zmloukc **VIRTUALBOX** SDK aznrqhv
xlhzm **VALGRIND** eunvzatrd **ANTLR** sob **BASH**
p **UBUNTU** iqnbuhysqsb **KDE** gyfoczdvsaivc
gqumdvce **l**df **TCPIP** xqvznd **VISUALSTUDIO**
vr **PHONON** kilte **lozc** **CgXML** pvsnecqyclhf
nlcbxur **il** **ACE** idpbaqz **bwko** **.NET** pfcx **GDB**
i **GPG** geskxhoq **JAVASCRIPT** vmvdxe **QML** zvd
khare **CMAKE** rcowwxqv **volmc** **RUBY** xlrh **GCC**
HTML luxgspi **PUBLICKEY** duevdp **OPENGL** tr
vmve **WT** s **GIT** cfweiasdmni **LIBVIRT** abs **CSS**
leuhdoksmd **WINDOWS** useshamh **lqu** **MAC** cri
GSTREAMER **lhdix** **laq** **SUBVERSION** olys **bC#**

Temas

1 Yo

2 Qt

3 Javascript

Introducción

Sentencias

Comentarios

Variables

Operadores

Comparaciones

Condicionales

Funciones

Bucles: for, while, break

Objetos

Cadenas

Fechas

Vectores

Booleanos

Matemáticas

Qt

¿Qué es Qt?

Framework multiplataforma

Para aplicaciones con y sin interfaz gráfica

Usado en todos los ámbitos (escritorio, web* y empotrado)

Licenciado bajo LGPL en 2009

Gratis para todo el mundo: <http://qt.nokia.com/>

Qt

Plataformas

Oficial

Windows

Mac

X11 (Linux...)

Symbian

Meego

Win Mobile

VxWorks

QNX

Android (4.8)

No oficial

iPhone

Kindle

Palm webOS

Bada

OS/2

Haiku-BeOS

NO Windows Phone 7 (!!!)



Qt

¿Por qué Qt?

Qt es intuitiva

Un único código para todas las plataformas

Tres licencias para adecuarse a tus necesidades (Comercial, LGPL o GPL)

Gran soporte de la comunidad

Hay herramientas gratuitas para empezar a aprender

Usada por 250.000 desarrolladores (comercial y software libre)



Algunas cosas sobre Qt

Aplicaciones Qt

Autodesk software (Maya y 3D Studio)

Google Earth

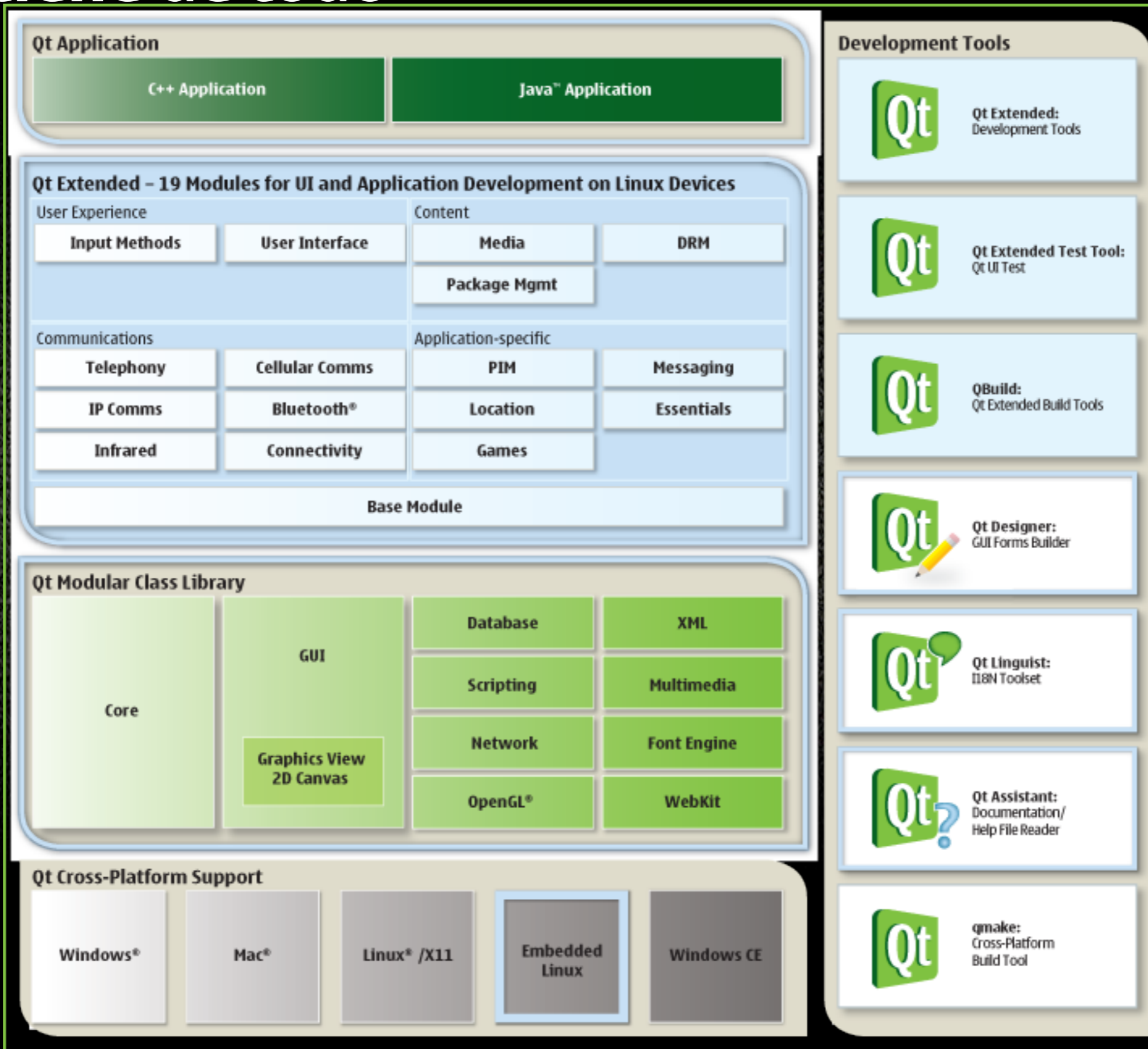
Skype para Linux

Opera

Soporte completo para dispositivos Nokia

Algunas cosas sobre Qt

Qt tiene de todo



Temas

1 Yo

2 Qt

3 Javascript

Introducción

Sentencias

Comentarios

Variables

Operadores

Comparaciones

Condicionales

Funciones

Bucles: for, while, break

Objetos

Cadenas

Fechas

Vectores

Booleanos

Matemáticas

Introducción

Creado por Netscape en 1995 para páginas web

EL lenguaje de programación en web

Parece un juguete, es un arma de destrucción masiva

Herencia basada en prototipo

Orientado a objetos

Dinámico

Funciones de primer orden

Fuertemente integrado con QtQuick

Sentencias

Sensible a mayúsculas/minúsculas

No es necesario terminar las sentencias (OJO)

```
console.log("Hola mundo");
```


Comentarios

Igual que C99 / C++

Multilínea: `/* ... */`

Hasta final de línea: `//`

En .qml, lo mismo

```
/*  
  Esto es un comentario  
  de dos líneas  
*/  
  
// Esto es un comentario de una línea
```


Variables

Contenedores de un único dato

Se usan para guardar valores para el futuro

Ese dato puede ser simple (p. ej. un número) o complejo (p. ej. número + texto + imagen + ...)

Se definen con 'var'

```
var x;
```


Operadores

Aritméticos

Operator	Description	Example	Result	
+	Addition	$x=y+2$	$x=7$	$y=5$
-	Subtraction	$x=y-2$	$x=3$	$y=5$
*	Multiplication	$x=y*2$	$x=10$	$y=5$
/	Division	$x=y/2$	$x=2.5$	$y=5$
%	Modulus (division remainder)	$x=y\%2$	$x=1$	$y=5$
++	Increment	$x=++y$	$x=6$	$y=6$
		$x=y++$	$x=5$	$y=6$
--	Decrement	$x=--y$	$x=4$	$y=4$
		$x=y--$	$x=5$	$y=4$

Asignación

Operator	Example	Same As	Result
=	$x=y$		$x=5$
+=	$x+=y$	$x=x+y$	$x=15$
-=	$x-=y$	$x=x-y$	$x=5$
=	$x=y$	$x=x*y$	$x=50$
/=	$x/=y$	$x=x/y$	$x=2$
%=	$x\%=y$	$x=x\%y$	$x=0$

Comparaciones

Ojo a == vs ===

Operator	Description	Example
==	is equal to	x==8 is false x==5 is true
===	is exactly equal to (value and type)	x===5 is true x==="5" is false
!=	is not equal	x!=8 is true
>	is greater than	x>8 is false
<	is less than	x<8 is true
>=	is greater than or equal to	x>=8 is false
<=	is less than or equal to	x<=8 is true

Condicionales

if sentencia

if ... else sentencia

if ... else if ... else sentencia

```
var x = 1;
```

```
if ( x === 2 ) {  
  foo();  
} else {  
  bar();  
}
```

switch

```
var x = 1;
```

```
switch ( x ) {  
  case 1: foo1(); break;  
  case 2: foo2(); break;  
  case 3: foo3(); break;  
  default: bar(); break;  
}
```


Funciones

Permiten agrupar varias sentencias y reusarlas

Imprescindibles para código limpio

```
function hola()
{
  console.log("Hola mundo");
}

hola();
```

En QtQuick se usan para la lógica de la aplicación y acceso a datos

Bucles

Repeticiones

Varios tipos según cuándo se hacen las comprobaciones

Nunca:

```
var i;  
  
for ( i = 0; i <= 5; i++) {  
    console.log("El valor de i es " + i);  
}
```

Antes:

```
var i = 0;  
  
while ( i <= 5 ) {  
    console.log("El valor de i es " + i);  
    i++;  
}
```

'break' rompe el bucle

```
var i;  
  
for ( i = 0; i <= 5; i++) {  
    console.log("El valor de i es " + i);  
    if ( i === 3) {  
        break;  
    }  
}
```

'continue' se salta valores

```
var i;  
  
for ( i = 0; i <= 5; i++) {  
    console.log("El valor de i es " + i);  
    if ( i === 3) {  
        break;  
    }  
}
```


Objetos

Tipos simples: enteros, flotantes, booleanos, null, NaN

Objeto: tipo compuesto, puede tener funciones propias (métodos)

Ejemplos de objetos: cadenas, fechas, vectores, los que definamos, etc

```
persona = new Object();  
persona.nombre = "Perico";  
persona.apellidos = "Palotes";  
persona.edad = 50;
```

Se pueden recorrer con for:

```
var i;  
  
for ( x in persona ) {  
    console.log(persona[x] + " ");  
}
```


Cadenas

Textos

Se pueden concatenar con +

```
var nombre = "Perico";  
var apellidos = "Palotes";  
var nombreCompleto = nombre + apellidos;
```


Fechas

El objeto Date permite obtener fecha, hora, año, etc

```
var hoy = new Date();  
hoy.getDay();  
hoy.getTime();
```

```
var manana = new Date();  
manana = hoy + 1;
```

Y mucho más...

Vectores

Array = variable que contiene más de un valor al mismo tiempo

```
var juego1 = "Portal";  
var juego2 = "World of Warcraft";  
var juego3 = "Angry Birds";
```

VS

```
var juegos = new Array();  
juegos[0] = "Portal";  
juegos[1] = "World of Warcraft";  
juegos[2] = "Angry Birds";
```


Booleanos

Sólo pueden tener 2 valores: cierto o falso

Falso: 0, -0, null, "", false, undefined, NaN

Cierto: todo lo que no es falso

Matemáticas

Operaciones y constantes matemáticas de unos habitual

Math.PI

Math.sqrt(valor)

Math.round(valor)

Math.random()

Math.max(valor1, valor2)

Math.min(valor1, valor2)

Math.E

Math.SQRT2

Math.LN2

....