

CONFIG += c++11

1 Slide 2.4

Introduction to QtCreator. Can show creating a new project, advertise Qt as a cross platform framework (with many extension to c++), mobile android/ios... games+applications.

```
#include<iostream>

int main()
{
    std::cout << "Hello ,_world!" << std::endl;
    return 0;
}
```

2 Slide 2.5

```
c++ -c main.cpp
okteta main.o
c++ -o tddd86 main.o
okteta tddd86
```

3 Slide 2.6

Show the effect of return value.

4 Slide 2.10

```
int a[100];
int* p = a;
a[0] = 12;
std::cout << a[0] << "_ " << p[0] << std::endl;
int b = 10;
int& p;
int& p = b;
std::cout << b << "_ " << p << std::endl;
b = 12;
std::cout << b << "_ " << p << std::endl;
```

5 Slide 2.11

do some pointer arithmetic with p (set some more value in a);

6 Slide 2.12

show the similitude between "int* const" and "int&".

7 Slide 2.14

show “using namespace std;”.

8 Slide 2.15

Show functions:

```
int max(int a, int b)
{
    return (a < b) ? b : a;
}
```

```
struct point
{
    int x, y;
    int max()
    {
        return max(x, y);
    }
};
```

// Then show public/private

9 Slide 2.17

Switch the declaration between main and max

10 Slide 2.18

Add the prototype.

11 Slide 2.22

Show the swap function in action

12 Slide 2.23

Show returning more than two values

```
void minmax(int a, int b, int& min, int& max)
{
    ...
}
```

Large object

```

struct t
{
    int a[100];
};

void this_copy(t v)
{
    std::cout << v.a << std::endl;
}

void this_doesnot_copy(t& v)
{
    std::cout << v.a << std::endl;
}

int main(int argc, char** argv)
{
    t t1;
    std::cout << t1.a << std::endl;
    this_copy(t1);
    this_doesnot_copy(t1);
}

```

13 Slide 2.24

Show the compilation error

```

void grow(const int& age) {
    age = age + 1;
}

int main() {
    int age = 20;
    grow(age);
    return 0;
}

```

14 Slide 2.30

Show some string comparison

```

string s1 = "apa";
string s2 = "apa";
std::cout << (s1 > s2) << std::endl;

s1 = "apaw";
s2 = "apa";

```

```

std::cout << (s1 > s2) << std::endl;

s1 = "apaw";
s2 = "apb";
std::cout << (s1 > s2) << std::endl;

std::cout << ("apaw" > "apa") << std::endl;

```

15 Slide 2.31

```

const char* c_string = "apa";
std::string cpp_string = cstring;
const char* other_c_string = cpp_string.c_str();

```

16 Slide 2.32

```

void printDouble(string s) {
cout << s * 2 << endl;
}

```

```

void appendFour(int n) {
cout << n + "4" << endl;
}

```

```

int main() {
    appendFour(2);
}

```

Then in `appendFour` replace "4" by "1234"

17 Slide 2.36

```

string name;
cout << "Type your name: "; // Type your name: %\underbar{John Doe}%
cin >> name;
cout << "Hello, " << name << endl; // Hello, John

```

Replace `cin` with `"getline(cin, name);"`