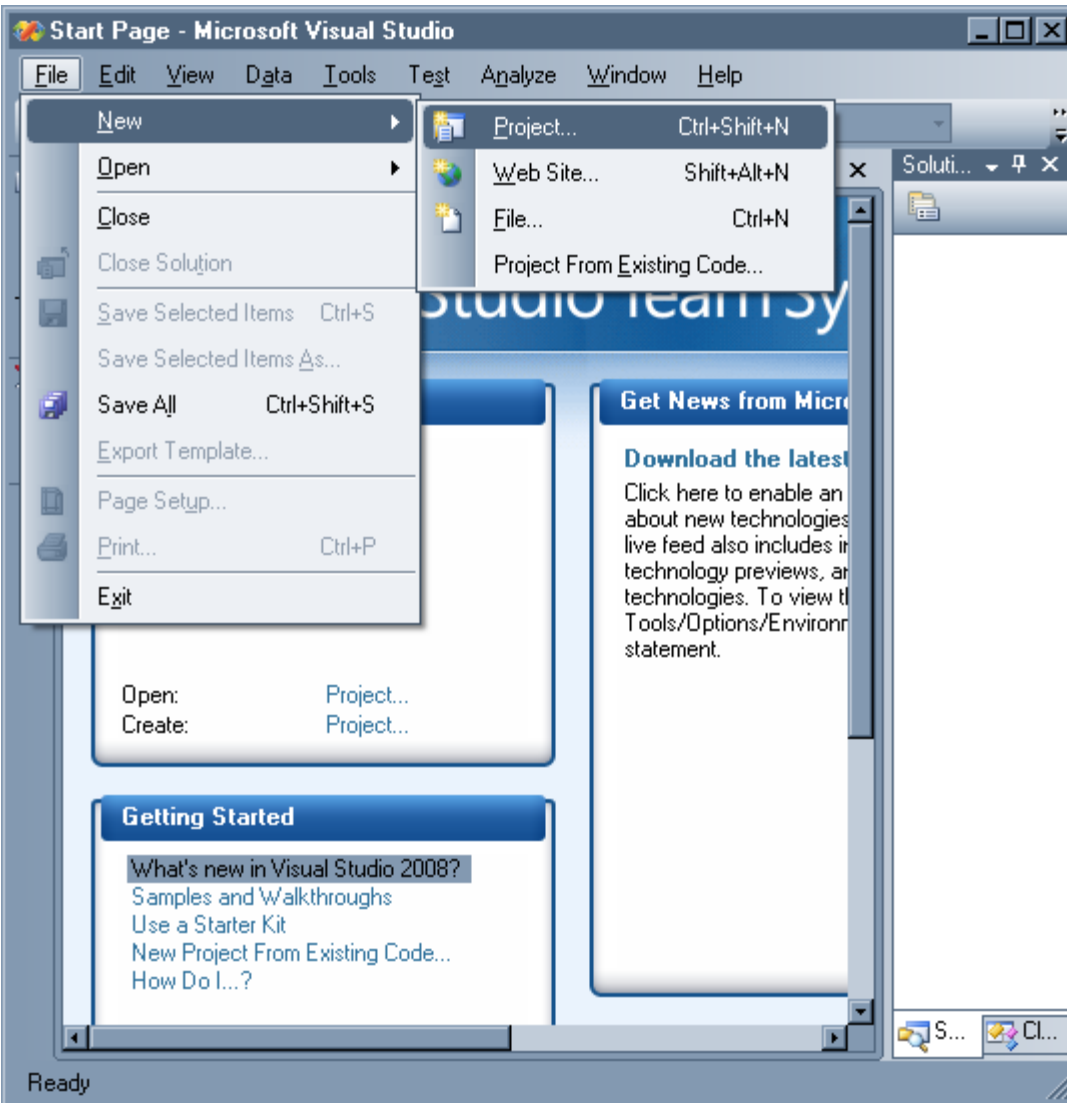




Dasar Pemrograman

- **Visual Studio 2008**
- **Program C++ Sederhana**

- C++ merupakan perkembangan leboh lanjut dari C. Bahasa C dikembangkan tahun 1972 oleh *Dennis Richie* di Bell Telephone laboratories.
- Oleh karena C tidak memiliki kemampuan OOP (Object Oriented Programming), dikembangkanlah C++ yang merupakan versi C yang mendukung OOP
- C++ mulai dikembangkan tahun 1979 oleh Bjarne Stroustrup di Bell Labs sebagai kelanjutan dari bahasa C.

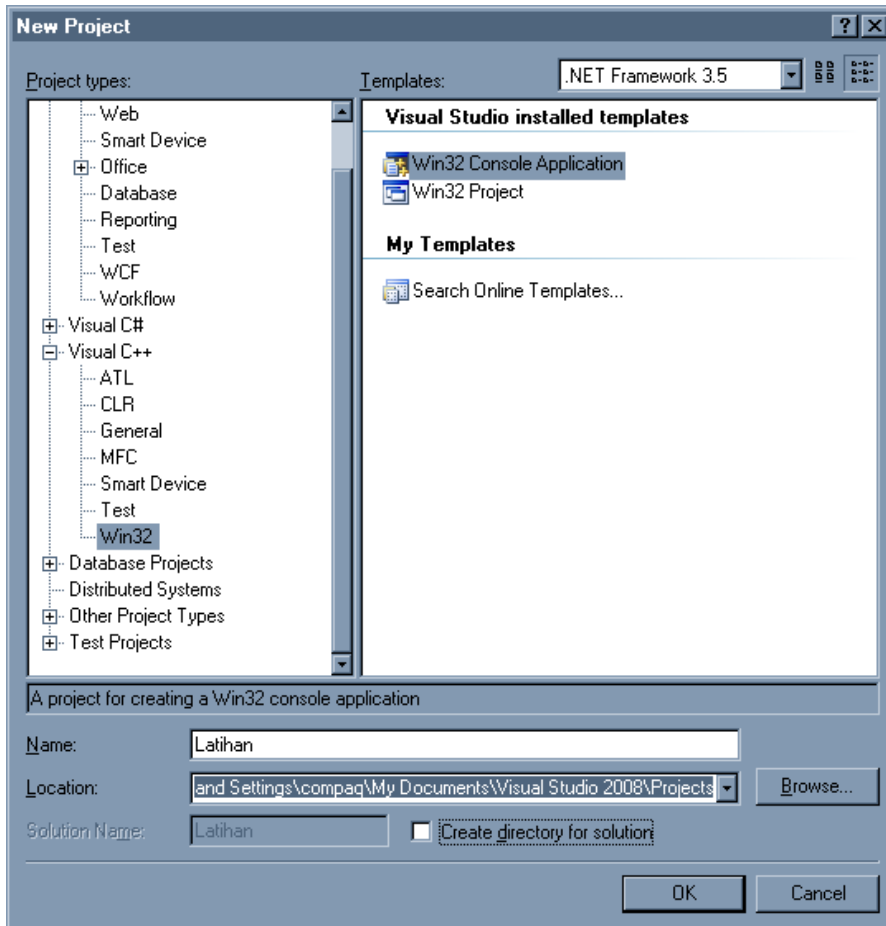


Memulai Project :

File → New →
Project

(atau tekan
CTRL + Shift +N)

Visual Studio 2008

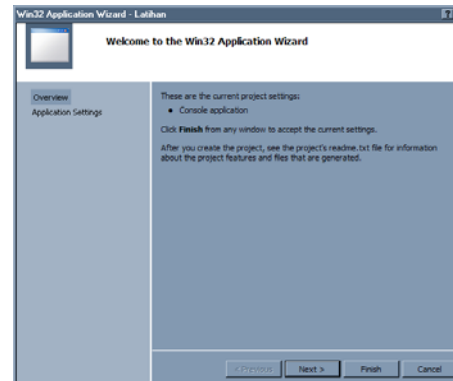


Project types :

- Pilih Visual C++ → Win32

Templates :

- Pilih Win32 Console Application
- Tulis nama Project
- Lepaskan tanda centang pada *Create Directory for Solution*



Klik OK,

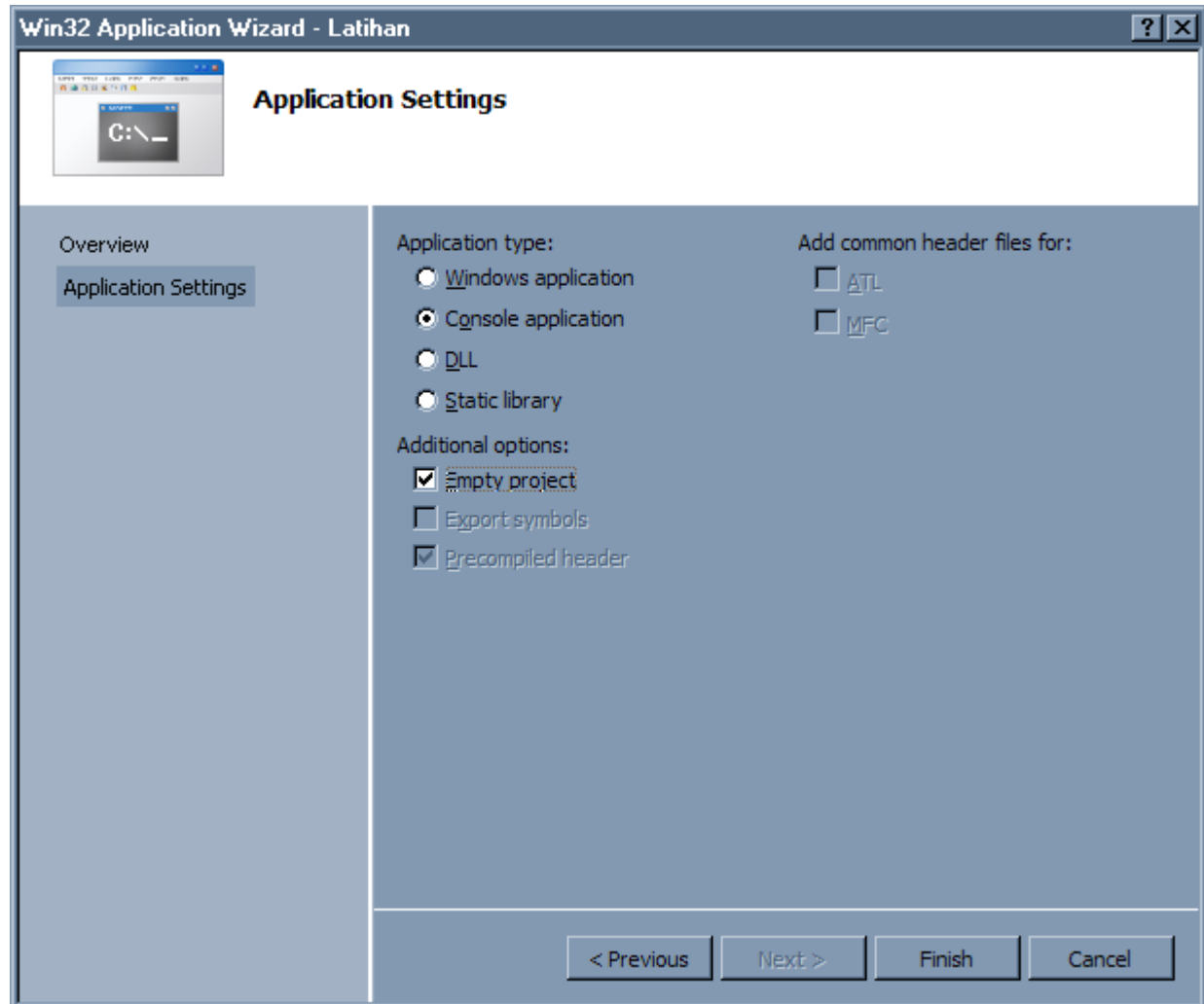
Akan tampil Win32 Application Wizard, Klik **Next**

Application type :

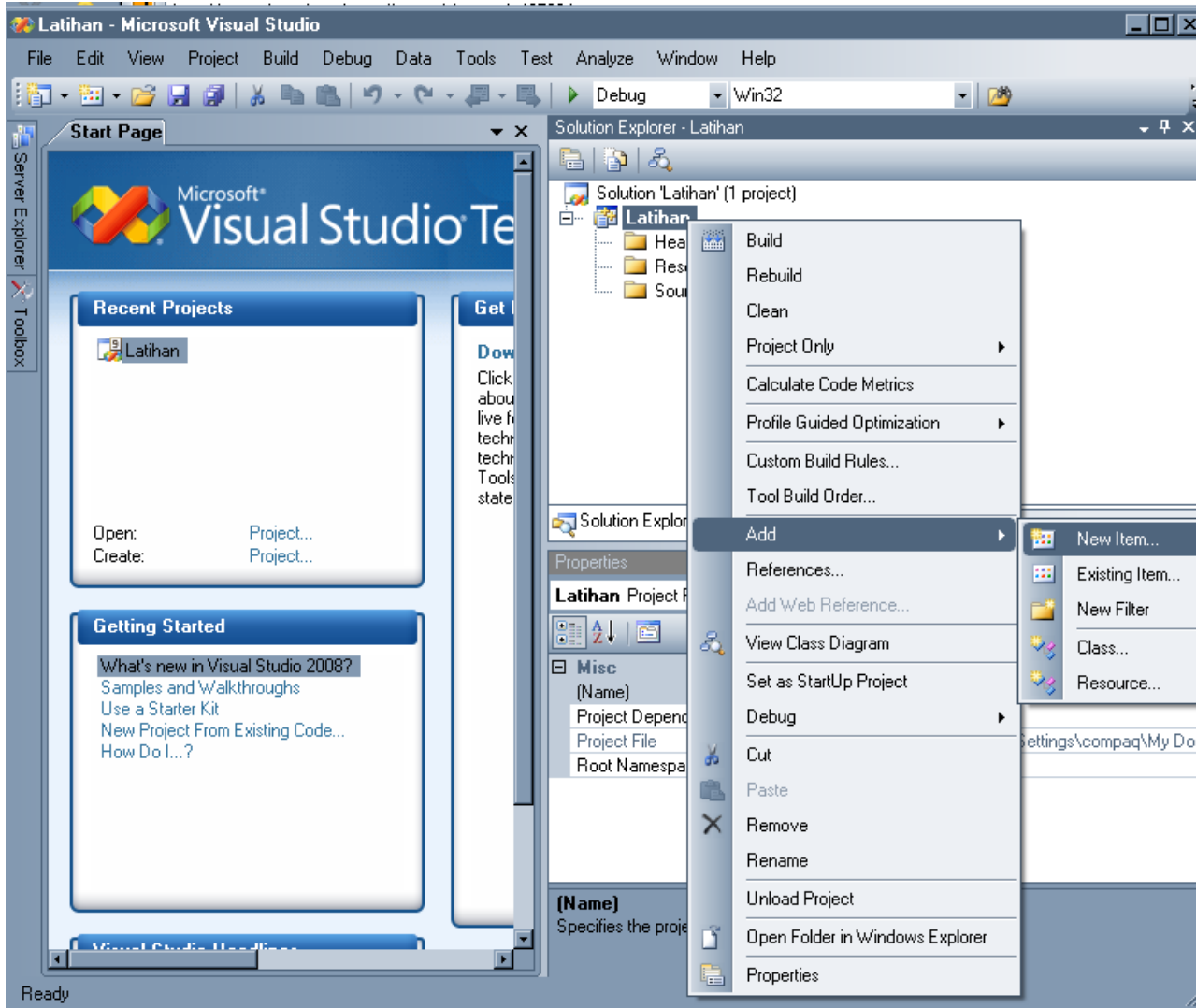
- pilih *Console application*

Additional options :

- Pilih *Empty Project*
- Klik *Finish*

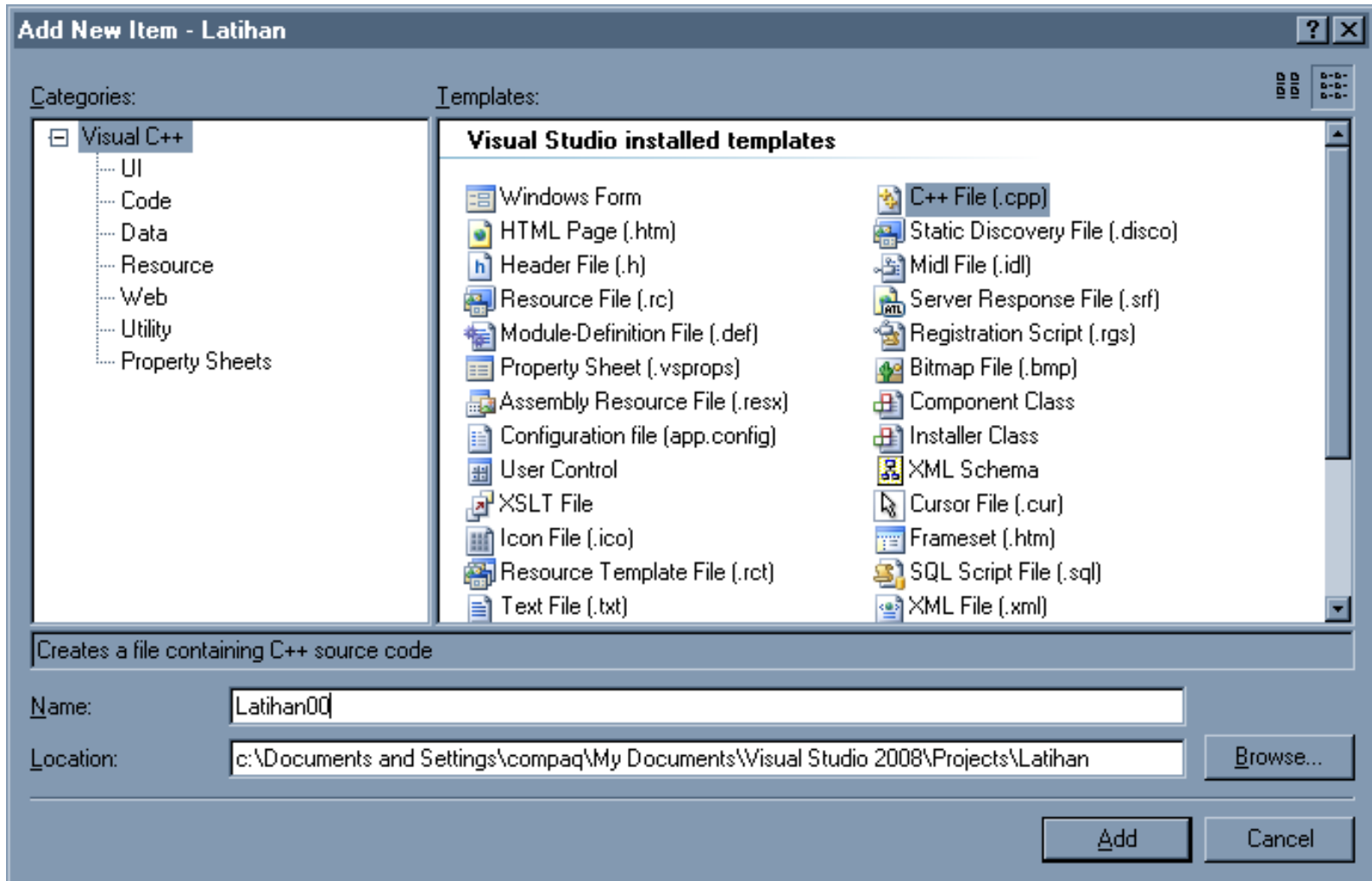


Visual Studio 2008

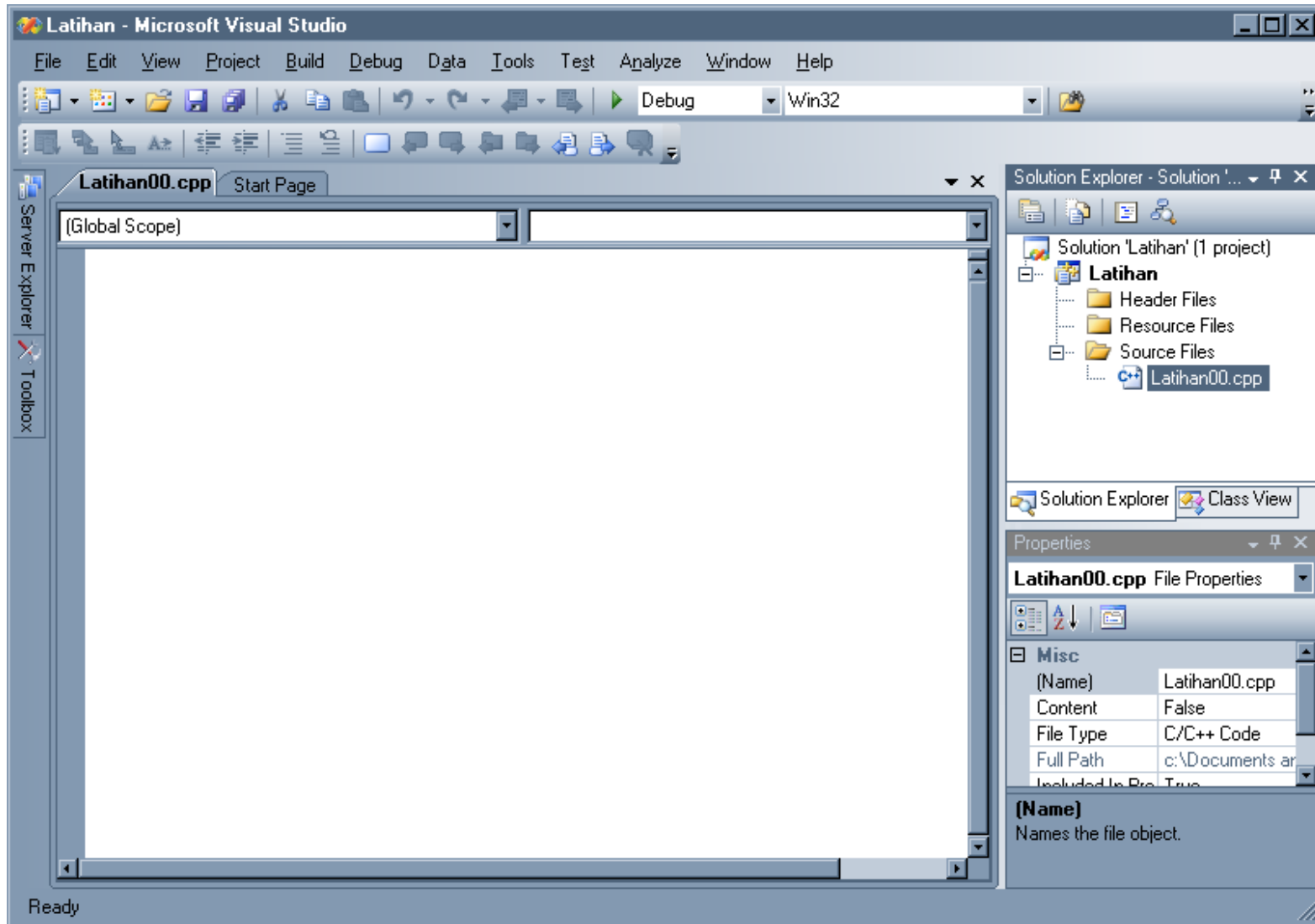


- Pada Solution Explorer, klik kanan nama Project → pilih Add → New Item

Visual Studio 2008



Visual Studio 2008



Program C++

- Program C++ terdiri dari objek, fungsi, variabel, dan bagian-bagian lain.
- Contoh sederhana program C++ (1) :

```

1: #include <iostream>
2: using namespace std;
3: void main()
4: {
5:     cout << "Hello World!\n";
6: }
    
```

\n digunakan agar cursor berpindah ke baris baru

\n dapat diganti dengan *cout <<endl;*

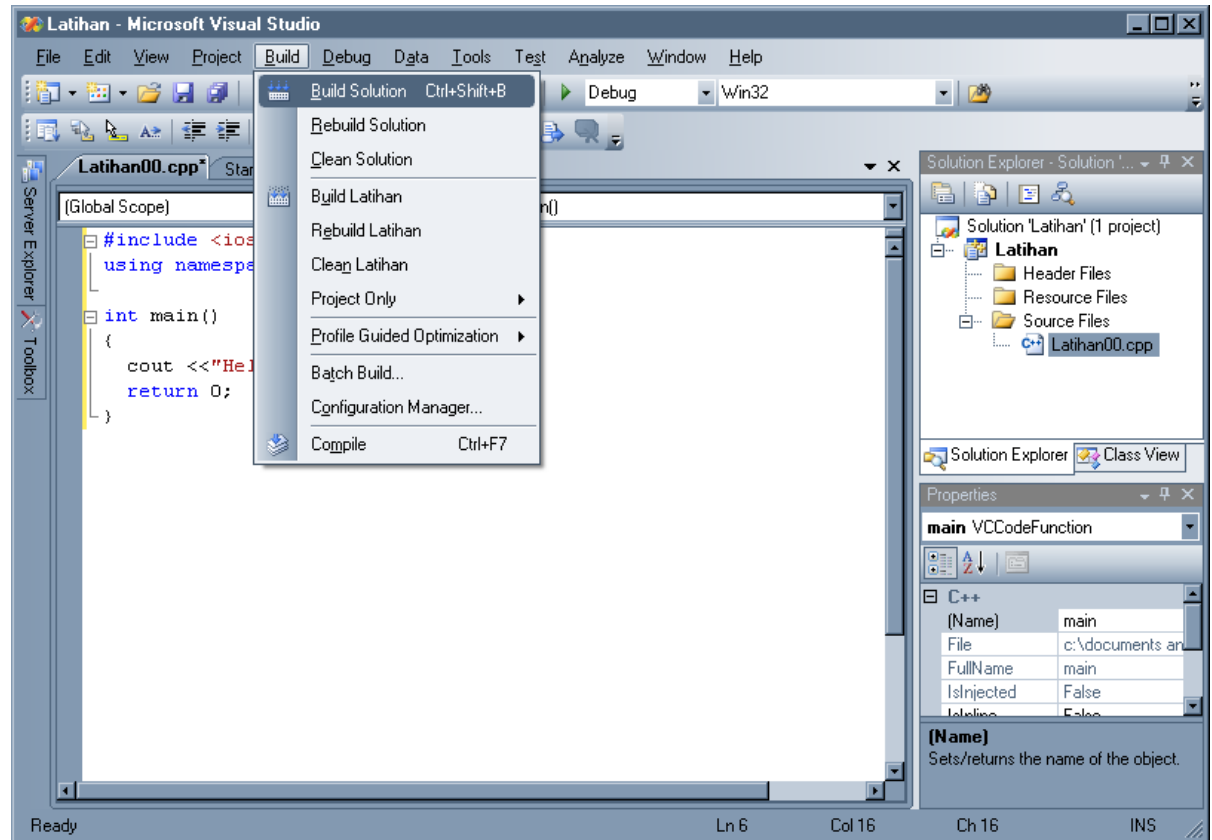
Contoh :

cout<<"hello world!"<<endl;

Hello World!

Program tersebut akan mencetak Hello World! ke layar

- Pilih menu Build → Build Solution kemudian
- Tekan F5 (atau pilih menu Debug → Start Debugging)



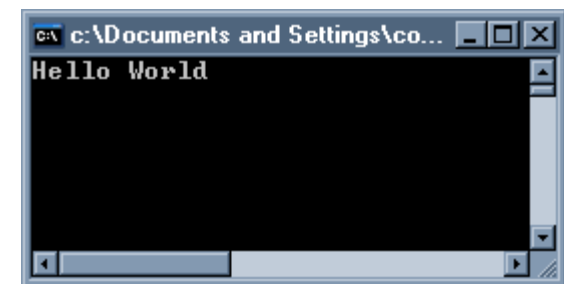
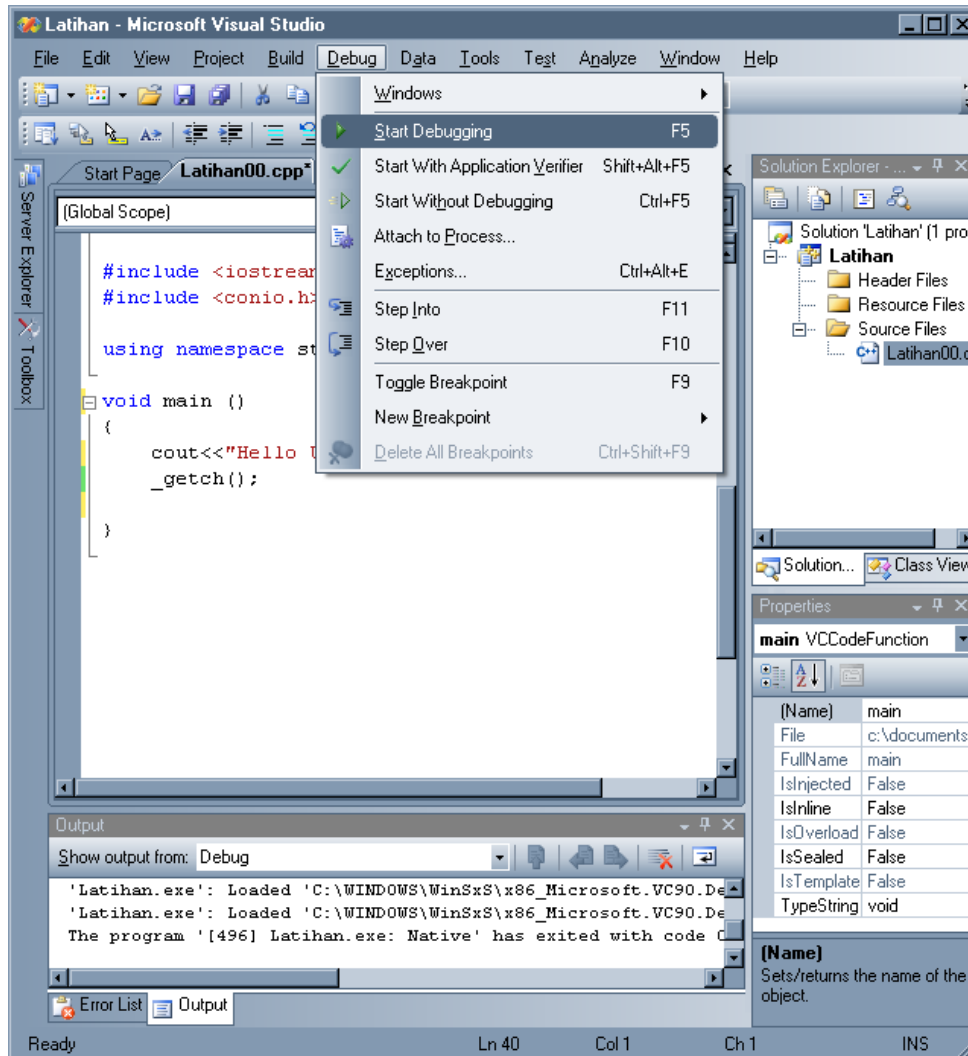
Program C++

- Contoh sederhana program C++ (2) :

```
1: #include <iostream>
2: #include <conio.h>
3: using namespace std;
4: void main()
5: {
6:     cout << "Hello World!\n";
7:     _getch();
8: }
```

Hello World!

Program tersebut akan mencetak Hello World!
ke layar



#include <iostream>

- Baris yang dimulai dengan tanda (#) merupakan preprocessor.
- Kode tersebut mengindikasikan *compiler's preprocessor*.
- Bentuk umum preprocessor bisa berupa :
 - 1) #include <namafile>
 - 2) #include "namafile"
- Bentuk pertama mengisyaratkan bahwa pencarian file dilakukan pada direktori khusus. Sedangkan bentuk kedua menyatakan bahwa pencarian file (*namafile*) dilakukan pertama kali pada direktori kerja.
- Jika hendak menggunakan file header yang disediakan oleh C++ maka gunakan bentuk pertama. Sedangkan jika memakai file header yang diciptakan sendiri maka akan menggunakan bentuk #include yang kedua

#include <iostream>

- Preprocessor disebut juga pengarah kompiler yang berfungsi mengatur proses kompilasi.
- Dalam contoh #include <iostream> dan #include <conio.h>, preprocessor tersebut menginstruksikan kompiler untuk membaca kode program yang ada dalam file `iostream` dan `conio`.
- #include <iostream> meminta preprocessor untuk menyertakan *iostream standard file*.
 - file (`iostream`) tersebut mencakup deklarasi-deklarasi standar untuk **input-output** library pada C++.
- file (`iostream`) disertakan karena fungsi-fungsi dalam file (`iostream`) tersebut akan digunakan kemudian di program.

using namespace std;

- Semua elemen dari library standar C++ dideklarasikan dengan **namespace std**.
- Baris tersebut sangat sering dicantumkan di program-program C++ karena kita membutuhkan fungsi-fungsi dari library standar tersebut.

main ()

- Baris ini berhubungan dengan definisi awal dari main function pada program.
- Main function merupakan titik awal bagi semua program C++ untuk mengeksekusi programnya.
- Kata main diikuti dengan kurung buka dan kurung tutup () yang merupakan indikasi bahwa main adalah sebuah fungsi.

(suatu fungsi dalam C++ harus mempunyai tanda “()”)

- Suatu fungsi jg diikuti dengan tanda { }, sebagai awal dan akhir suatu fungsi.

Apa yang terdapat dalam tanda { } tersebut adalah statement-statement yang akan dieksekusi ketika fungsi tersebut dipanggil.

main()

Void main()

- Tidak memiliki nilai balik

```
Void main ()  
{ //statements  
}
```

int main()

- Memiliki nilai balik, pernyataan return harus ditambahkan, untuk menyatakan nilai balik

```
Int main ()  
{ //statements  
  return 0 ;  
}
```

cout << "Hello World!";

- Baris tersebut merupakan statement C++.
- Suatu statement adalah ekspresi sederhana maupun kompleks yang dapat menghasilkan suatu efek.
- Cout merepresentasikan standar output stream pada C++, dan arti dari `cout << "Hello World!";` tersebut adalah untuk memasukkan rangkaian karakter (dalam hal ini tulisan Hello World!) ke dalam standard output stream (biasanya berupa layar).
- `cout` dideklarasikan dalam `iostream` standard file termasuk juga `std` namespace, oleh karena itulah kita harus meng-`#include` file tersebut dan mendeklarasikan namespace tersebut di awal kode.

```
cout << "Hello World!";
```

- statement tersebut diakhiri dengan karakter semicolon (;).
- Karakter tersebut digunakan untuk menandai akhir dari suatu statement dan HARUS selalu digunakan di akhir semua *expression statement* di program C++.
- Kesalahan programmer pemula yang paling umum adalah lupa menyertakan karakter ‘;’ tersebut di akhir suatu baris.

return 0;

- Return merupakan contoh statement yang menginstruksikan agar suatu fungsi mengembalikan nilai balik tertentu.
- Return membuat suatu fungsi berakhir.
- Return dapat diikuti dengan suatu *return code* . Seperti contoh, return code 0 untuk fungsi main() mempunyai arti bahwa program bekerja seperti yang diharapkan, tanpa error selama dieksekusi.
- Biasanya return 0; digunakan untuk mengakhiri suatu C++ console program.