# Façade – Фасада

**Фасада** (на [английски](http://bg.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8_%D0%B5%D0%B7%D0%B8%D0%BA): *Facade*) е структурен [шаблон за дизайн](http://bg.wikipedia.org/wiki/%D0%A8%D0%B0%D0%B1%D0%BB%D0%BE%D0%BD%D0%B8_%D0%B7%D0%B0_%D0%B4%D0%B8%D0%B7%D0%B0%D0%B9%D0%BD), който се използва в [обектно-ориентираното](http://bg.wikipedia.org/wiki/%D0%9E%D0%B1%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%B0%D0%BD%D0%BE_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B8%D1%80%D0%B0%D0%BD%D0%B5) [програмиране](http://bg.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B8%D1%80%D0%B0%D0%BD%D0%B5).

Предоставя уеднаквен интерфейс за редица интерфейси. Фасадата дафинира интерфейс от по-високо ниво, което прави по-лесна употребата на подсистамата.

/\* Complex parts \*/

class CPU {

public void Freeze() { ... }

public void Jump(long position) { ... }

public void Execute() { ... }

}

class Memory {

public void Load(long position, byte[] data) { ... }

}

class HardDrive {

public byte[] Read(long lba, int size) { ... }

}

/\* Facade \*/

class ComputerFacade {

private CPU processor;

private Memory ram;

private HardDrive hd;

public ComputerFacade() {

this.processor = new CPU();

this.ram = new Memory();

this.hd = new HardDrive();

}

public void Start() {

processor.Freeze();

ram.Load(BOOT\_ADDRESS, hd.read(BOOT\_SECTOR, SECTOR\_SIZE));

processor.Jump(BOOT\_ADDRESS);

processor.Execute();

}

}

/\* Client \*/

class You {

public static void Main() {

ComputerFacade computer = new ComputerFacade();

computer.Start();

}

}

