Российский университет дружбы народов
Институт гостиничного бизнеса и туризма

В. Дихтяр

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

(для бакалавров)

Раздел 1. Разработка алгоритмов и программ

Тема 1-5. *Концепция объектно-ориентированного программирования*

Москва 2016

Базовые понятия

- ***Объект Ô***
- свойство ***Ô***
- метод обработки
- событие
- класс ***Ô = cls Ô***

Ô— совокупность свойств определенных сущностей и методов их обработки (***ПЭ***-средств).

Свойство — характеристика (параметр) ***Ô***.

{***Ô*** }наделены определенными свойствами, которые в совокупности выделяют ***Ô*** из множества других ***Ô***.

Пример

- ***Ô :*** перечисление свойств:

  ***Ô*_A (*свойство-1, свойство-2, …,свойство-k*).**

- Свойства ***Ô*** различных ***cls*** могут «пересекаться» ⇒ возможны ***Ô***, обладающие одинаковыми свойствами:

  ***Ô*_B (...*свойство-n, свойство-m,...свойство-r,*...)**

  ***Ô*_C (...*свойство-n,…, свойство-r,*...)**.

Абстракция (1)

≡ способность отображать ***Ô*** внешнего мира в форме абстрактных ***Ŝ*** (структур) в соответствии с решаемой задачей.

Абстрактные структуры, при помощи которых реализуется этот принцип, называются ***cls***.

cls ≡ структура, описывающая ***Ô*** внешнего мира с помощью двух типов элементов:

состояние ***Ô*** описывается полями класса (переменными разного типа), а

поведение ***Ô***— его методами (процедурами и функциями).

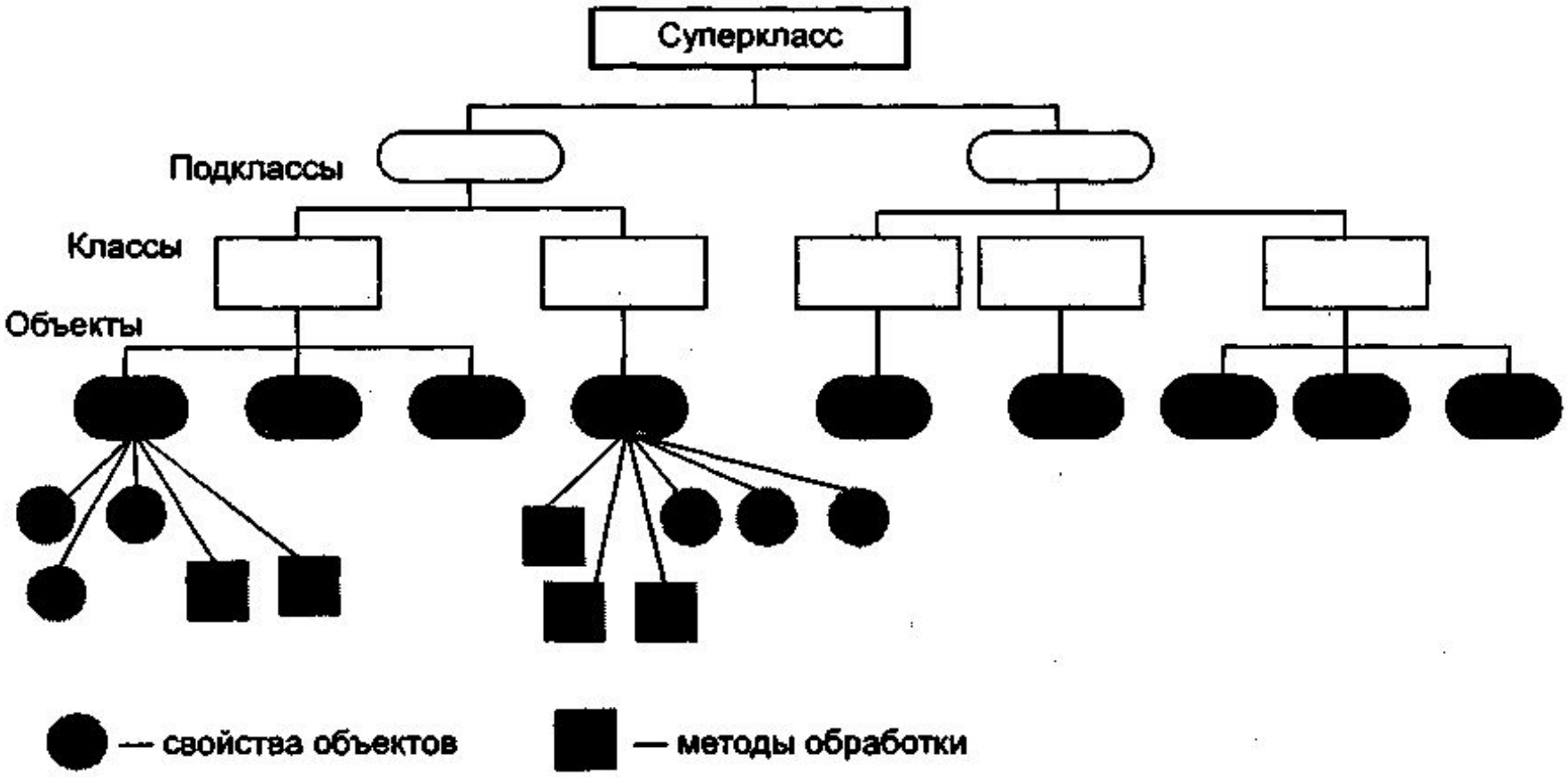
Абстракция (2)

cls - абстрактные описания структур ***Ď***, но сами ***Ď*** они не содержат.

Ď появляются тогда, когда по описаниям ***cls*** выделяется необходимое пространство и в нем создаются *экземпляры* ***cls***, или ***Ô***.

Тогда для каждого поля ***cls*** отводится необходимая область памяти и в эту область можно поместить нужное значение.

Связь основных понятий ООР



Методика объектно-ориентированного проектирования

Объектно-ориентированные ***τ*** и методики проектирования ***ɳ***-продуктов (обеспечение выполнения принципов ***Ô***-подхода):

- инкапсуляция (замыкание) свойств ***Ď*** и ***ɳ*** в ***Ô***;
- наследование;
- полиморфизм

Инкапсуляция = сочетание ***ŜĎ*** с методами их обработки в абстрактных типах ***Ď - clsÔ***

Полиморфизм — способность ***Ô*** реагировать на запрос сообразно своему типу (одно и то же имя метода может использоваться для различных ***cls Ô***)

Свойства методик объектно-ориентированного проектирования

- ***Ô*** описывается как ***Θ*** некоторой сущности реального мира;
- ***Ô***, для которых определены места хранения, рассматриваются во взаимосвязи, и применительно к ним создаются ***ҧ-****модули Š*.

Объектно-ориентированный анализ

- идентификация ***Ô*** и их свойств;
- перечень ***Q***, выполняемых над каждым ***Ô***, в зависимости от его состояния;
- связи между ***Ô*** для образования ***cls***;
- требования к интерфейсу с ***Ô.***

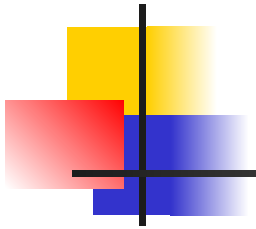
4 этапа объектно-ориентированного проектирования

- разработка диаграммы аппаратных средств системы обработки ***Ď***, показывающей процессоры, внешние устройства, вычислительные сети и их соединения;
- разработка структуры ***cls***, описывающей связь между ***cls*** и ***Ô***;
- разработка диаграмм ***Ô***, показывающих взаимосвязи с другими ***Ô***;
- разработка внутренней ***Ŝ*** (***ҍ***-продукта)

Концепция объектно-ориентированного программирования

Основные принципы:

- абстракция,
- наследование,
- инкапсуляция,
- полиморфизм.



Наследование

≡ свойство ***cls*** порождать другие ***cls***ы таким образом, что в порожденном ***cls*** (***cls***-потомке) содержатся все поля и все методы ***cls*** - родителя (базового ***cls***), а также дополнительно собственные поля и методы.

Пример 1

Измерительные приборы: термометр, барометр и весы.

Чтобы описать эти приборы внутри ***ҧ***, создается три ***cls***

cls **«термометр»**	***cls*****«весы»**	***cls*****«барометр»**
Инв. номер	Инв. номер	Инв. номер
Название	Название	Название
Цвет	Цвет	Цвет
Исполнение	Исполнение	Исполнение
Температура	Вес	Давление
Изменить температуру	Изменить вес	Изменить давление

Поля классов

Метод классов

Пример 1

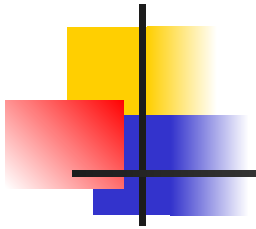
Методы ***cls*** представляют реализацию некоторых действий - описывают поведение ***cls***.

Например, метод ***cls*** «термометр», который носит название «измерить температуру», призван помещать значение температуры в поле с именем «температура».

Экземпляр класса «термометр»		**Экземпляр класса класс «весы»**		**Экземпляр класса класс «барометр»**	
Инв. номер	1	Инв. номер	3	Инв. номер	3
Название	TRM	Название	VSV	Название	DRM
Цвет	белый	Цвет	серый	Цвет	хром
Исполнение	оконный	Исполнение	напольный	Исполнение	настенный
Температура	10	Вес	2	Давление	230
Изменить температуру		Изменить вес		Изменить давление	

Пример 2

У разных ***cls*** есть одинаковые поля: Инв. номер. Название, Цвет, Исполнение.
Наследование: класс Измеритель, в котором повторяющиеся поля,
а затем ***cls*** Термометр, Весы и Барометр



Инкапсуляция (1)

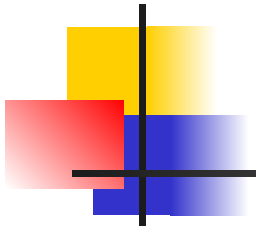
≡ способность ***cls*** скрывать от внешнего мира детали внутренней реализации.

Области видимости обозначаются в структуре ***cls*** специальными служебными словами:

private - все поля и методы, объявленные в этой области видимости, невидимы для внешнего мира,

public - все, объявленное в этой зоне, для внешнего мира открыто.

Инкапсуляция (2)



У ***cls*** появляется защитный корпус, предохраняющий от излишней перегрузки ***ъ*** объявленными именами и методами.

Наружу ***cls*** экспонирует только то, что составляет его интерфейс и предназначено для внешнего взаимодействия.

Все остальные поля и методы он скрывает внутри, помещая их в область видимости private.

Полиморфизм

≡ способность ***Ô***ов выполнять одну и ту же команду разными способами.

Свойства:

- способности трактовать ***Ô cls***-наследника как тип базового ***cls***
- способности ***cls*** иметь виртуальные методы.

Особенность виртуального метода - надежная связь с созданным экземпляром ***cls***.

Пример

Если добавить в каждый из ***cls***ов (в класс Измеритель и каждый из его наследников) метод Сигналить, а в реализации этого метода описали бы для каждого ***cls*** свой способ подавать сигнал (для термометра — мигать, для весов — издавать звук сирены, а для барометра — свистеть), то мы могли бы всем ***Ô***ам, потомкам класса Измеритель, дать команду: измерители, сигналить!

И каждый из них подал бы сигнал: термометры бы замигали, весы издали звук сирены, а барометры засвистели. Это и есть результат работы виртуальных методов.