

# C++ , Object Pascal

1

www.algopars.ir

*Object Pascal C++*

*Object Oriented*

( )  
(Case Study)

*Object Pascal OO*

*Object Pascal C++*

C++ , Object Pascal , Delphi , OOP:

.1

2

(OOA/OOD)

(OOP)

Object C++

Borland Delphi

Pascal

C++

Object Pascal

---

aboazar@algopars.ir<sup>1</sup>  
Software Development<sup>2</sup>

1

(Visual)

.2

Object Model

OO

6 5 4 3

Object

Heap, Stack

6

Plain Object Model

Object Reference Model

Heap

7

object

Plain Object Model C++

Object Reference Model Pascal

Pointer

( )

Plain Object Model

( )

Object Reference Model

Pointer

Object Reference Model

.3

Data Type

- 
- Abstraction<sup>3</sup>
  - Hiding<sup>4</sup>
  - Inheritance<sup>5</sup>
  - Polymorphism<sup>6</sup>
  - Reference<sup>7</sup>

Object Pascal C++

- 1. Myclass obj ;
- 2. Obj.MyMethod( );

Instance

Stack

Myclass

obj

MyMethod

obj

MyMethod

obj

Object Pascal

- 1. Var
- 2. obj: Myclass;
- 3. begin
- 4. obj := Myclass.Creat;
- 5. obj.MyMethod;

Myclass

C++

MyMethod

C++

obj .

C++

obj

Object Pascal

( )

<sup>8</sup>

Creat

obj

MyMethod

obj

begin

<sup>9</sup>

**.4**

---

Constructor <sup>8</sup>  
Syntax Error <sup>9</sup>

( Allocate )  
C++

Stack 10

Borland Delphi Object Pascal 11

( )

Owner

Owner

Check Box

... , Button

( Check Box , Button )  
TComponent

TCollection TStringList

7 (1) TComponent  
7

Owner 7

Object

---

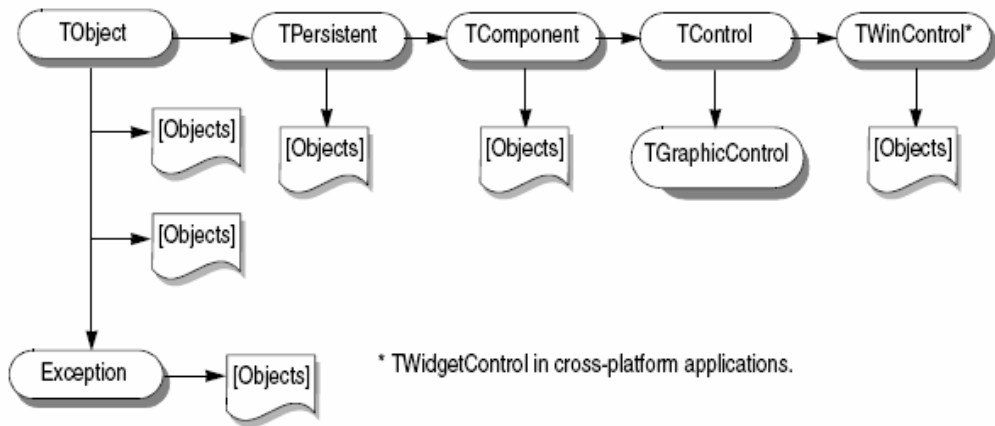
Destroy <sup>10</sup>  
Garbage <sup>11</sup>

.Net <sup>12</sup> 7  
 .Net Garbage Owner Collector  
 GC Garbage Collector  
 2006,2005,8

13

.Net  
 CLR<sup>14</sup>

.Net  
 Object Pascal C++  
 Garbage Collector



:(1)

15

.5

OO Constructor

Initialize

---

Platform <sup>12</sup>  
 Application <sup>13</sup>  
 Common Language Runtime <sup>14</sup>  
 Constructor and Destructor <sup>15</sup>

C++  
 C++  
 ( )  
 Constructor Object Pascal  
 ) TObject Creat  
 ( TObject  
 Object Pascal  
 Destructor  
 )  
 16 ( ... 17  
 Scope C++  
 Destructor  
 Destroy Object Pascal  
 Free Free  
 Owner  
 C++

18 **.6**

---

Object Life Time <sup>16</sup>  
 Window Handles <sup>17</sup>  
 Class Encapsulation <sup>18</sup>

Private , Protected , Public

OOP

19

- : Public
  - : Protected
  - : Private
- C++

Private  
C++

by pass  
C++

Friend  
Friend

private protected

Private protected

Object Pascal  
20

Automated  
RTTI <sup>21</sup>

Published  
OLE Automation

Published  
Automated

**.7**

C++ Object Pascal  
22

Object Pascal C++  
C++

h )  
23

( CPP

- 
- Visibility <sup>19</sup>
  - Unit <sup>20</sup>
  - Run Time Type Information <sup>21</sup>
  - Header File <sup>22</sup>
  - Compilation <sup>23</sup>

Unit  
Implementation

Object Pascal  
Interface

Uses

**8**

OOP

Inheritance

26

25

Sub

Classing, Inheritance, Derivation

Derived Class, Child Class, Subclass

Bass Class, Parent Class, Super Class

Private, Protected, Public

C++

Dog

Public

Private

:

Animal

Class Dog: Public Animal{

....

};

Object Pascal

Object

Pascal

:

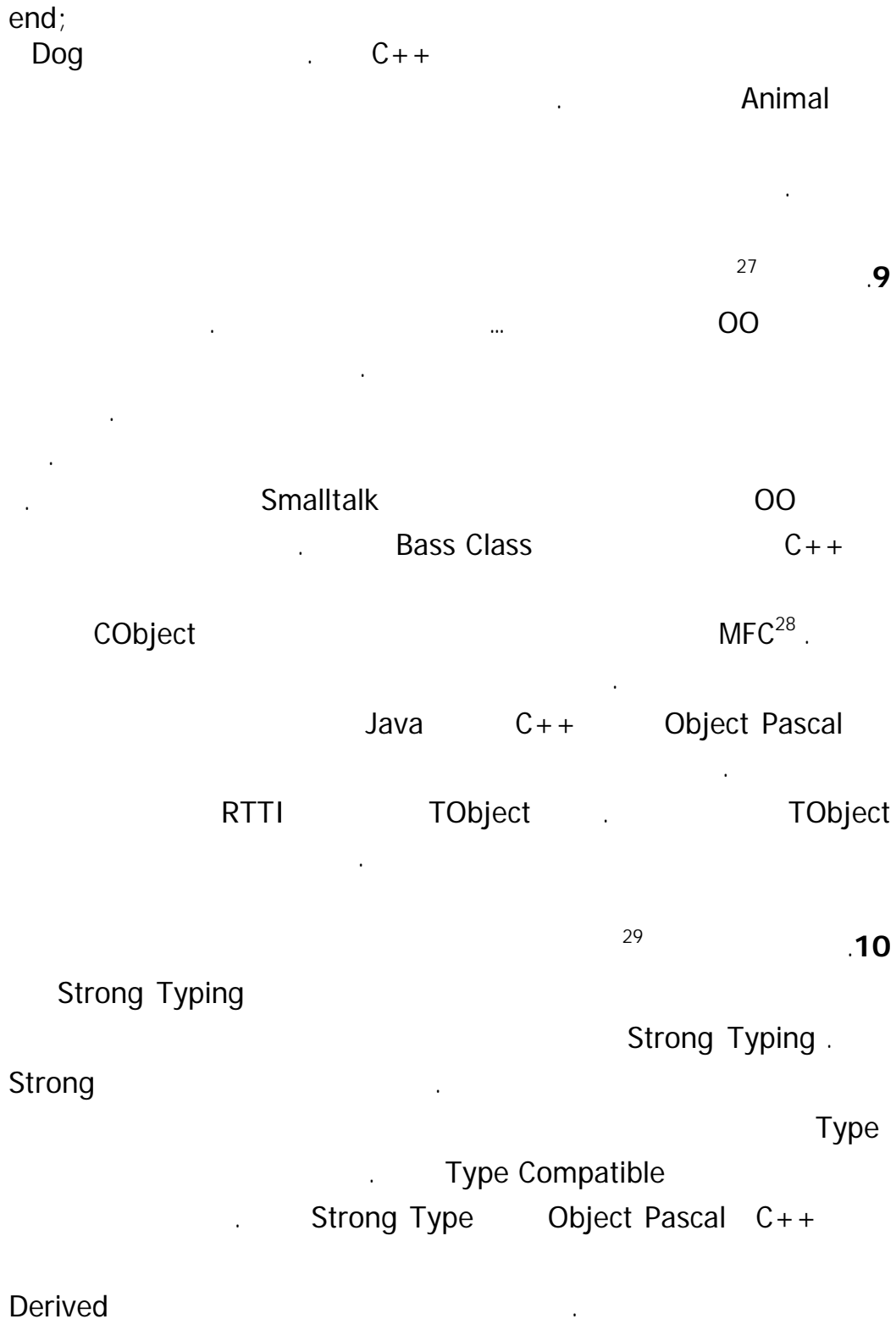
Type

Dog = class (Animal)

...

---

Linker<sup>24</sup>  
Generalization<sup>25</sup>  
Specialization<sup>26</sup>




---

Base Class<sup>27</sup>  
 Microsoft Fundamental Class<sup>28</sup>  
 Subtype Compatibility<sup>29</sup>

Bass class

Class

31

30

C++

Object Pascal

Object Reference Model

33

32

.11

Polymorphism

34

C++

Object Pascal

Virtual

Virtual Function

C++

Dynamic

35

.12

Abstraction

Bind

---

Pointers<sup>30</sup>  
 Reference<sup>31</sup>  
 Late Binding<sup>32</sup>  
 Polymorphism<sup>33</sup>  
 Abstraction<sup>34</sup>

Abstraction<sup>35</sup>

```

( ) ( ) ( )
) C++
Virtual
:
Virtual Void Fun_Name ( ) Const;
Fun_Name
C++
abstract Object Pascal
Object Pascal
Warning
36
37 .13
)
(
)
Object Pascal C++ (
Object Pascal
C++
38 C++

```

---

Runtime Error<sup>36</sup>  
Overloading<sup>37</sup>  
Type Safe Linkage<sup>38</sup>

Overload

39

Overload

40

**.14**

( )  
41

Object Pascal  
Interface

C++

Plain Object Model

Object Reference Model

Object

C++ Pascal

C++

Object Pascal C++

(OOP)

OOP

---

Exception<sup>39</sup>  
Multiple Inheritance<sup>40</sup>  
Interface<sup>41</sup>

1-Marco Cantu; Comparing OOP Languages: Java, Object Pascal, C++; Internet : [www.marcocantu.com](http://www.marcocantu.com)

2- Borland Software Corporation; Borland Delphi 7 Developer's Guide; Chapters : 3,4

3- Ray Lischner; Delphi in a Nutshell; O'Reilly & Associates Inc.; USA 2000; Chapter 2

4- Zarko Gajic; Fast Forward To Delphi For .Net; Internet : [www.delphi.about.com](http://www.delphi.about.com)

C++ : / 5

1379

: / 6

1379

