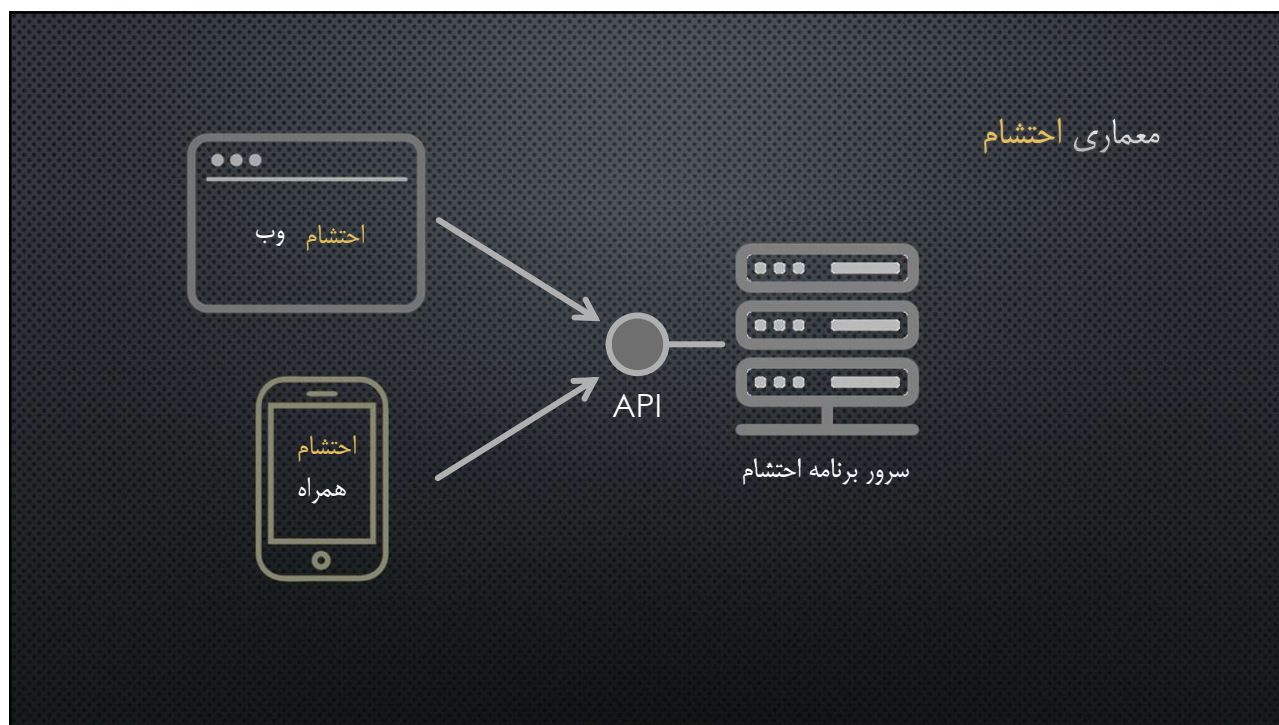


مقدمه‌ای بر طراحی شیء‌گرا

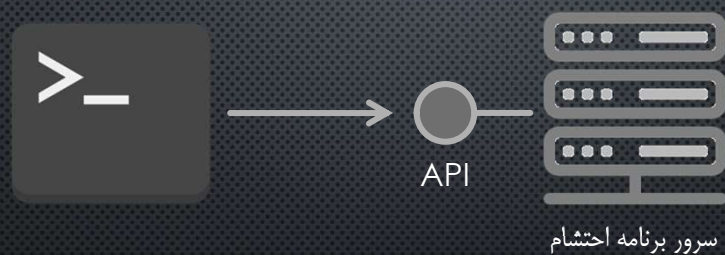
سامانه سفارش غذای **احتشام**

صورت مسئله

- کارمندان یک شرکت برای صرف ناهار از تعدادی مؤسسه تهیه غذای خارج شرکت غذا سفارش می‌دهند
- مشکل: ممکن است در یک روز چندین سفارش به یک مؤسسه داده شود -> تماس‌های مکرر و هزینه پیک زائد
- راه‌حل: طراحی یک سامانه برای تجميع سفارش‌های هر مؤسسه و تماس با هر مؤسسه توسط مسئول سفارش غذا
- فواید: صرف وقت کمتر از کارمندان و سرشکن شدن هزینه پیک



معماری احتشام



ورودی نمونه

request	1001	Farsi	Kabab
request	1002	Farsi	Kabab
request	1003	Alaa	Maahi
request	1002	Farsi	Jooj
catorder	Farsi		
catorder	Alaa		
delivery	Farsi	10000	
bedehi			
delivery	Alaa	10000	
bedehi			

ورودی نمونه

ثبت درخواست غذا

نام موسسه

request 1001 Farsi Kabab

شناسه کارمند

آیتم سفارش

ورودی نمونه

نمایش سفارش
موسسه

نام موسسه

catorder Farsi

ورودی نمونه

تحويل غذا به شرکت

هزینه پیک

delivery Farsi 10000

نام موسسه

ورودی نمونه

گزارش بدهی کارمندان

bedehi

ورودی نمونه

request	1001	Farsi	Kabab
request	1002	Farsi	Kabab
request	1003	Alaa	Maahi
request	1002	Farsi	Jooj
catorder	Farsi		
catorder	Alaa		
delivery	Farsi	10000	
bedehi			
delivery	Alaa	10000	
bedehi			

نمایش کلاس‌ها در نمادگذاری UML

Date
-day: int
-month: int
-year: int
+Date(day:int, month:int, year:int)
+set_date(day:int, month:int, year:int)
+print_date()

انعطاف پذیری نمایش کلاس‌ها

Date
-day -month -year
+Date() +set_date() +print_date()

انعطاف پذیری نمایش کلاس‌ها

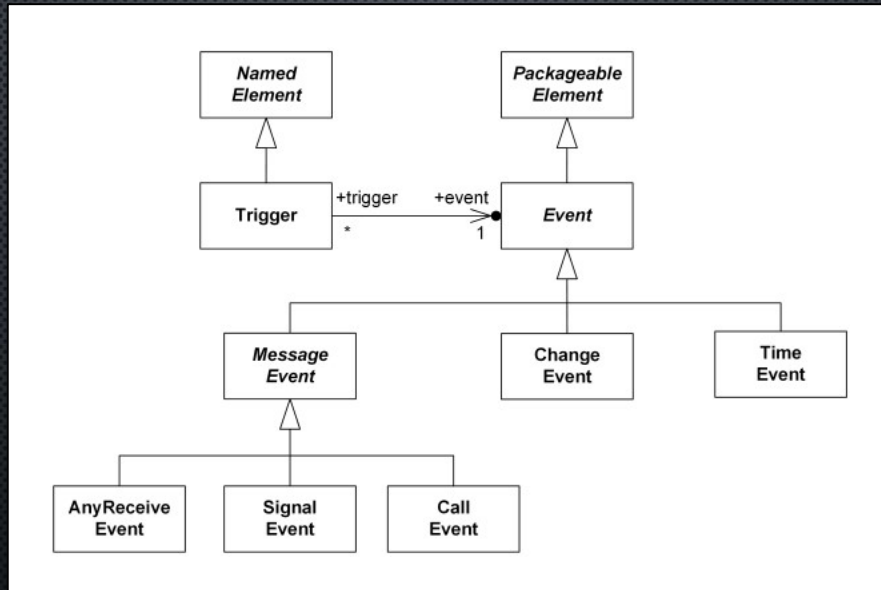
Date
-day -month -year
+set_date() +print_date()

انعطاف پذیری نمایش کلاس ها

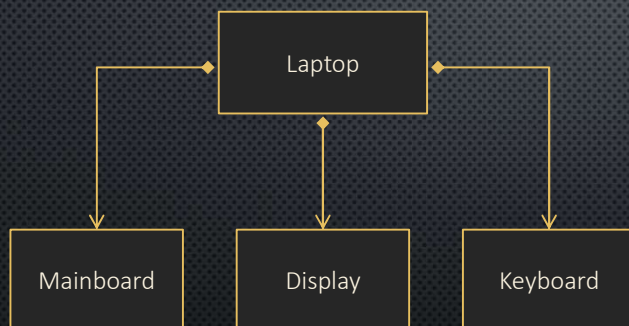
Date
-day
-month
-year

انعطاف پذیری نمایش کلاس ها

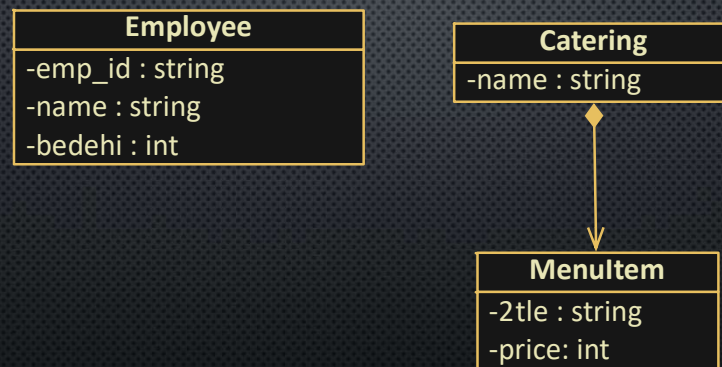
Date



رابطه composition (کل و جزء)

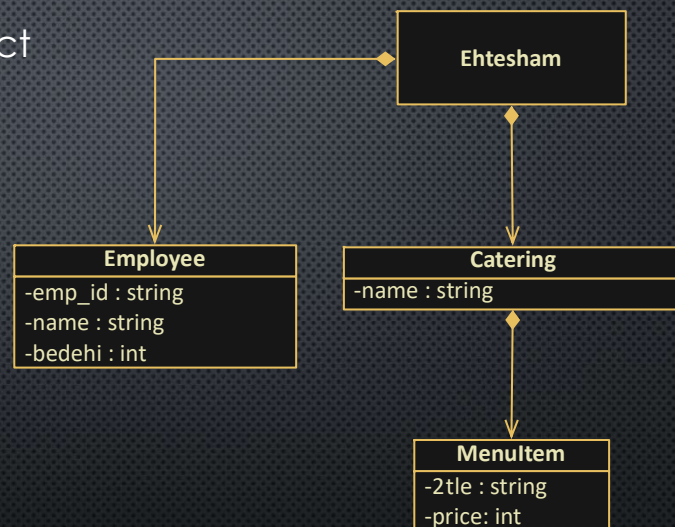


Basic classes in the domain



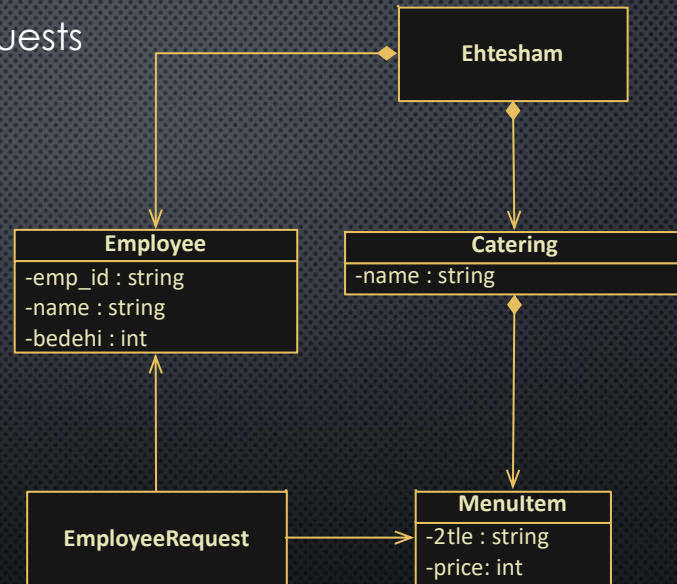
ehtesham1.cpp

Defining a top-level object

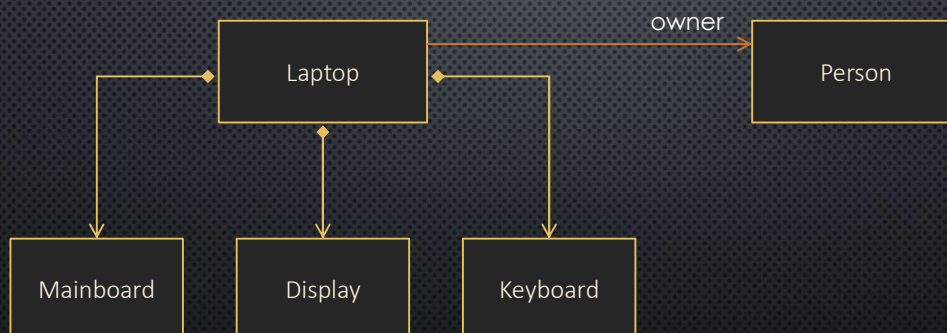


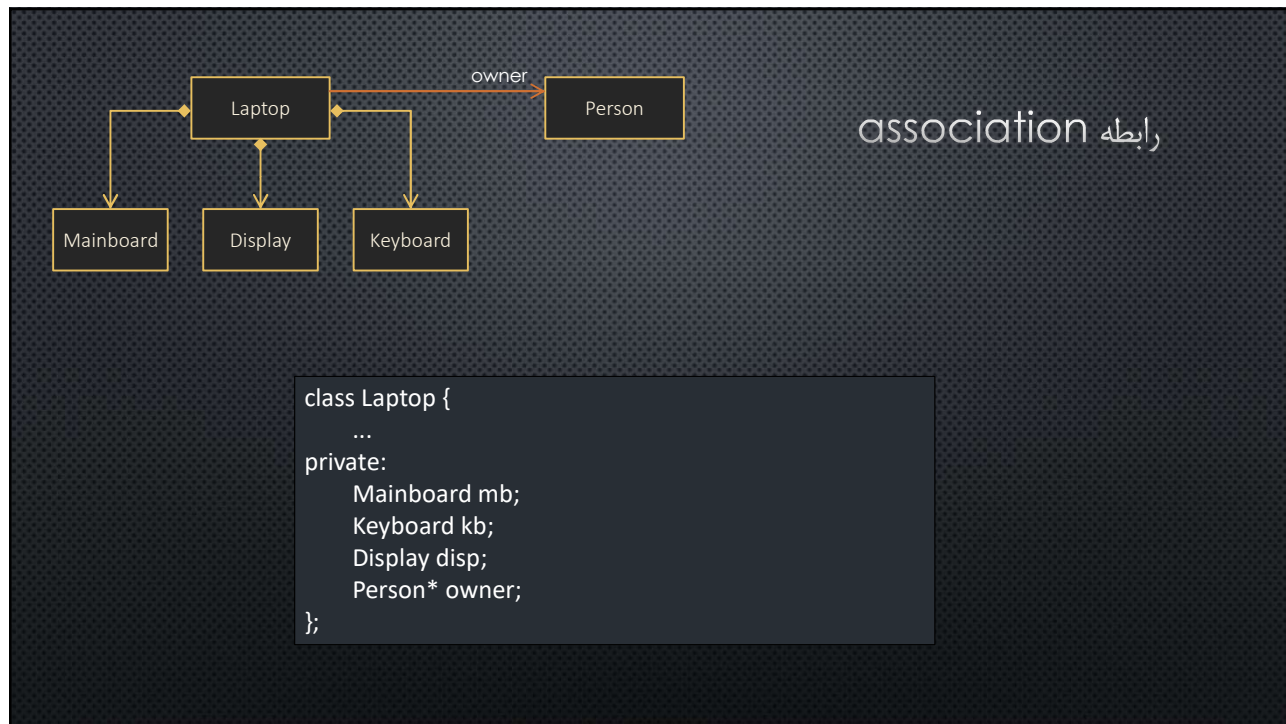
ehtesham2.cpp

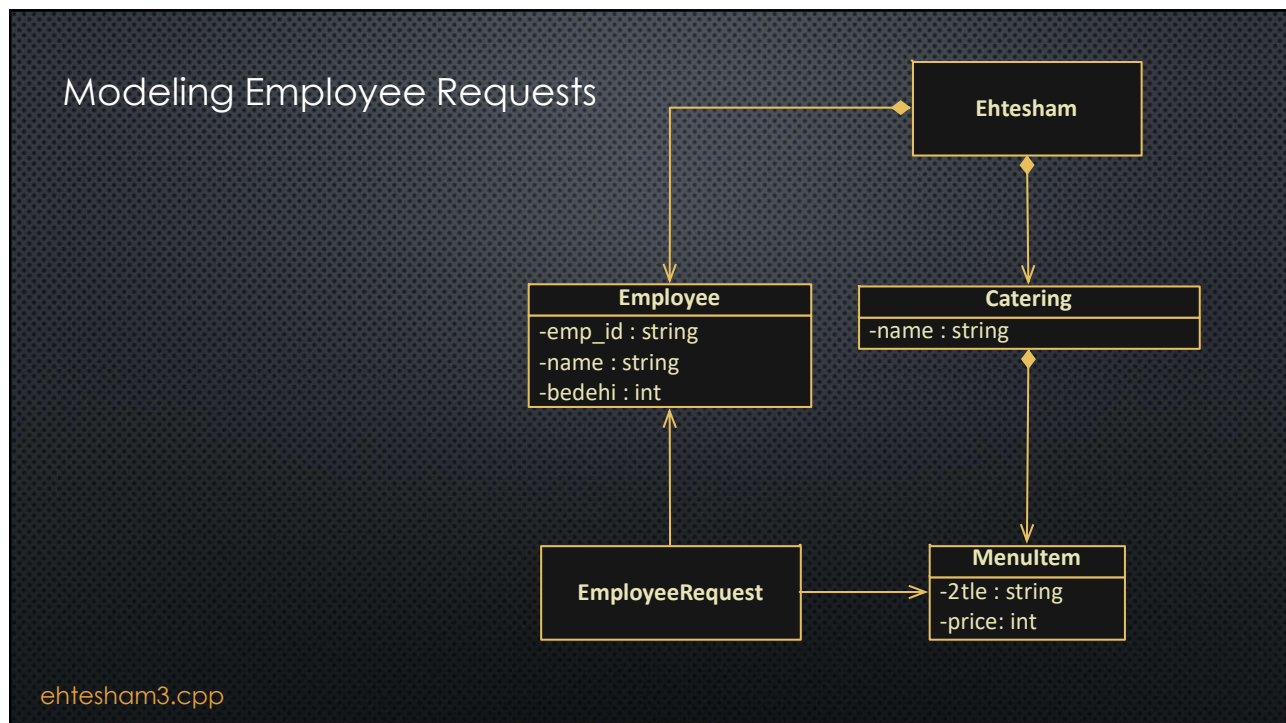
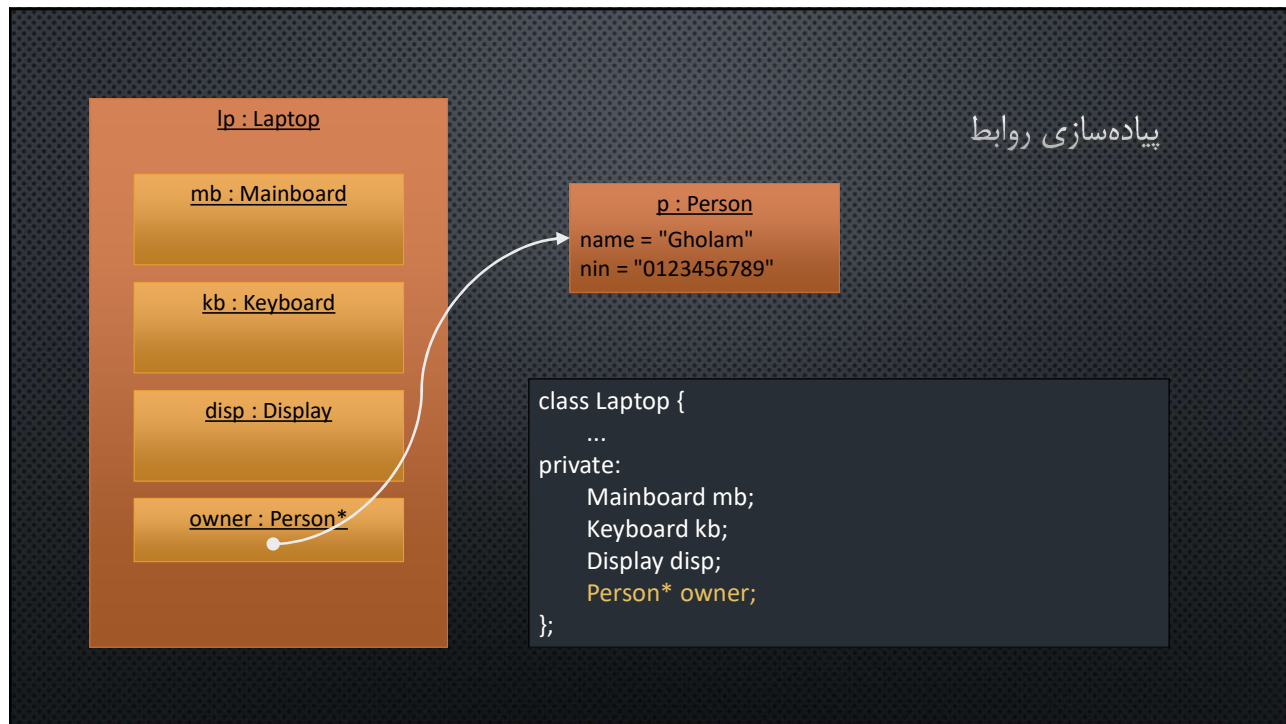
Modeling Employee Requests



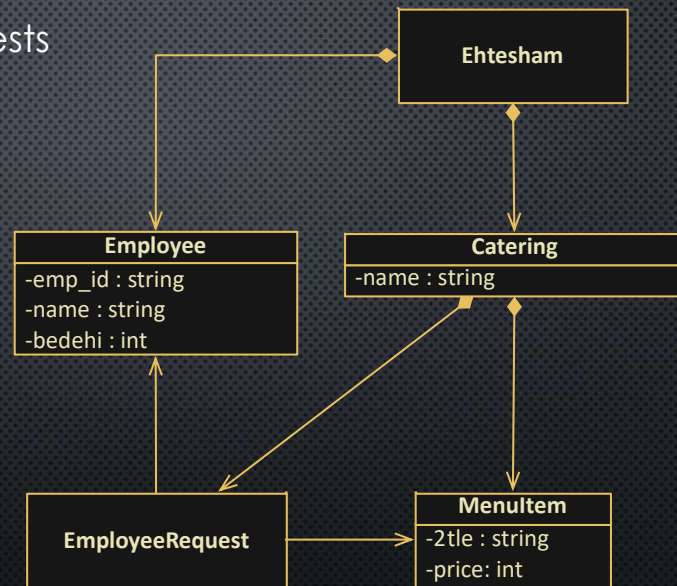
association رابطه





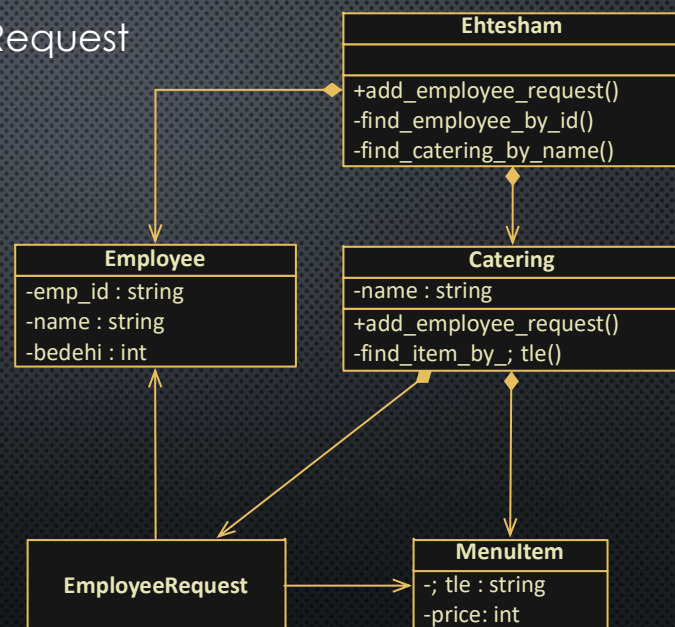


Holding Employee Requests



ehtesham4.cpp

Designing Add Employee Request



ehtesham5.cpp


```

void Ehtesham::add_employee_request(string empid, string catering_name, string item) {
    Catering cat = find_catering_by_name(catering_name);
    Employee emp = find_employee_by_id(empid);
    cat.add_employee_request(emp, item);
}

Catering Ehtesham::find_catering_by_name(string name) {
    for (int i = 0; i < caterings.size(); i++)
        if (caterings[i].get_name() == name)
            return caterings[i];
    error("cannot find catering");
}

```

```

void Ehtesham::add_employee_request(string empid, string catering_name, string item) {
    Catering cat = find_catering_by_name(catering_name);
    Employee emp = find_employee_by_id(empid);
    cat.add_employee_request(emp, item);
}

Catering Ehtesham::find_catering_by_name(string name) {
    for (int i = 0; i < caterings.size(); i++)
        if (caterings[i].get_name() == name)
            return caterings[i];
    error("cannot find catering");
}

```

ehtesham			
employees	1001	1002	1003
	"Jamshid"	"Mahvash"	"Hooshang"
	0	0	0
caterings	"Farsi"	"Ala"	
	{ ... }	{ ... }	
	{ }	{ }	

add_employee_request(...)	
item	"Kabab"
catname	"Farsi"
empid	1001
emp	emp_id
	name
	bedehi
cat	name
	menu
	requests

```

void Ehtesham::add_employee_request(string empid, string catering_name, string item) {
    Catering cat = find_catering_by_name(catering_name);
    Employee emp = find_employee_by_id(empid);
    cat.add_employee_request(emp, item);
}

Catering Ehtesham::find_catering_by_name(string name) {
    for (int i = 0; i < caterings.size(); i++)
        if (caterings[i].get_name() == name)
            return caterings[i];
    error("cannot find catering");
}

```

ehtesham			
employees	1001	1002	1003
	"Jamshid"	"Mahvash"	"Hooshang"
	0	0	0
caterings	"Farsi"	"Ala"	
	{ ... }	{ ... }	
	{ }	{ }	

find_catering_by_name("Farsi")

name

i

add_employee_request(...)

item

catname

empid

emp

emp_id	
name	
bedehi	

cat

name	
menu	
requests	

```

void Ehtesham::add_employee_request(string empid, string catering_name, string item) {
    Catering cat = find_catering_by_name(catering_name);
    Employee emp = find_employee_by_id(empid);
    cat.add_employee_request(emp, item);
}

Catering Ehtesham::find_catering_by_name(string name) {
    for (int i = 0; i < caterings.size(); i++)
        if (caterings[i].get_name() == name)
            return caterings[i];
    error("cannot find catering");
}

```

ehtesham			
employees	1001	1002	1003
	"Jamshid"	"Mahvash"	"Hooshang"
	0	0	0
caterings	"Farsi"	"Ala"	
	{ ... }	{ ... }	
	{ }	{ }	

find_catering_by_name("Farsi")

name

i

add_employee_request(...)

item

catname

empid

emp

emp_id	
name	
bedehi	

cat

name	
menu	
requests	


```

void Ehtesham::add_employee_request(string empid, string catering_name, string item) {
    Catering cat = find_catering_by_name(catering_name);
    Employee emp = find_employee_by_id(empid);
    cat.add_employee_request(emp, item);
}
Catering Ehtesham::find_catering_by_name(string name) {
    for (int i = 0; i < caterings.size(); i++)
        if (caterings[i].get_name() == name)
            return caterings[i];
    error("cannot find catering");
}

```

ehtesham

employees	1001	1002	1003
	"Jamshid"	"Mahvash"	"Hooshang"
	0	0	0

caterings	"Farsi"	"Ala"
	{ ... }	{ ... }
	{ }	{ }

find_catering_by_name("Farsi")

name	"Farsi"
i	0

add_employee_request(...)

item	"Kabab"						
catname	"Farsi"						
empid	1001						
emp	<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr><td style="padding: 2px 5px;">emp_id</td><td style="padding: 2px 5px;"></td></tr> <tr><td style="padding: 2px 5px;">name</td><td style="padding: 2px 5px;"></td></tr> <tr><td style="padding: 2px 5px;">bedehi</td><td style="padding: 2px 5px;"></td></tr> </table>	emp_id		name		bedehi	
emp_id							
name							
bedehi							
cat	<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr> <td style="padding: 2px 5px;">name</td> <td style="padding: 2px 5px;">"Farsi"</td> </tr> <tr> <td style="padding: 2px 5px;">menu</td> <td style="padding: 2px 5px;">{ ... }</td> </tr> <tr> <td style="padding: 2px 5px;">requests</td> <td style="padding: 2px 5px;">{ }</td> </tr> </table>	name	"Farsi"	menu	{ ... }	requests	{ }
name	"Farsi"						
menu	{ ... }						
requests	{ }						

```

void Ehtesham::add_employee_request(string empid, string catering_name, string item) {
    Catering cat = find_catering_by_name(catering_name);
    Employee emp = find_employee_by_id(empid);
    cat.add_employee_request(emp, item);
}
Catering Ehtesham::find_catering_by_name(string name) {
    for (int i = 0; i < caterings.size(); i++)
        if (caterings[i].get_name() == name)
            return caterings[i];
    error("cannot find catering");
}

```

ehtesham

employees	1001	1002	1003
	"Jamshid"	"Mahvash"	"Hooshang"
	0	0	0

caterings	"Farsi"	"Ala"
	{ ... }	{ ... }
	{ }	{ }

find_catering_by_name("Farsi")

name	"Farsi"
i	0

add_employee_request(...)

item	"Kabab"						
catname	"Farsi"						
empid	1001						
emp	<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr><td style="padding: 2px 5px;">emp_id</td><td style="padding: 2px 5px;"></td></tr> <tr><td style="padding: 2px 5px;">name</td><td style="padding: 2px 5px;"></td></tr> <tr><td style="padding: 2px 5px;">bedehi</td><td style="padding: 2px 5px;"></td></tr> </table>	emp_id		name		bedehi	
emp_id							
name							
bedehi							
cat	<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr> <td style="padding: 2px 5px;">name</td> <td style="padding: 2px 5px;">"Farsi"</td> </tr> <tr> <td style="padding: 2px 5px;">menu</td> <td style="padding: 2px 5px;">{ ... }</td> </tr> <tr> <td style="padding: 2px 5px;">requests</td> <td style="padding: 2px 5px;">{ <new req> }</td> </tr> </table>	name	"Farsi"	menu	{ ... }	requests	{ <new req> }
name	"Farsi"						
menu	{ ... }						
requests	{ <new req> }						

```

void Ehtesham::add_employee_request(string empid, string catering_name, string item) {
    Catering* cat = find_catering_by_name(catering_name);
    Employee* emp = find_employee_by_id(empid);
    cat->add_employee_request(emp, item);
}

Catering* Ehtesham::find_catering_by_name(string name) {
    for (int i = 0; i < caterings.size(); i++)
        if (caterings[i].get_name() == name)
            return &caterings[i];
    error("cannot find catering");
}

```

```

void Ehtesham::add_employee_request(string empid, string catering_name, string item) {
    Catering* cat = find_catering_by_name(catering_name);
    Employee* emp = find_employee_by_id(empid);
    cat->add_employee_request(emp, item);
}

Catering* Ehtesham::find_catering_by_name(string name) {
    for (int i = 0; i < caterings.size(); i++)
        if (caterings[i].get_name() == name)
            return &caterings[i];
    error("cannot find catering");
}

```

ehtesham			
employees	1001	1002	1003
	"Jamshid"	"Mahvash"	"Hooshang"
	0	0	0
caterings	"Farsi"	"Ala"	
	{ ... }	{ ... }	
	{ <new req> }	{ }	

find_catering_by_name("Farsi")

name "Farsi"

i 0

add_employee_request(...)

item "Kabab"

catname "Farsi"

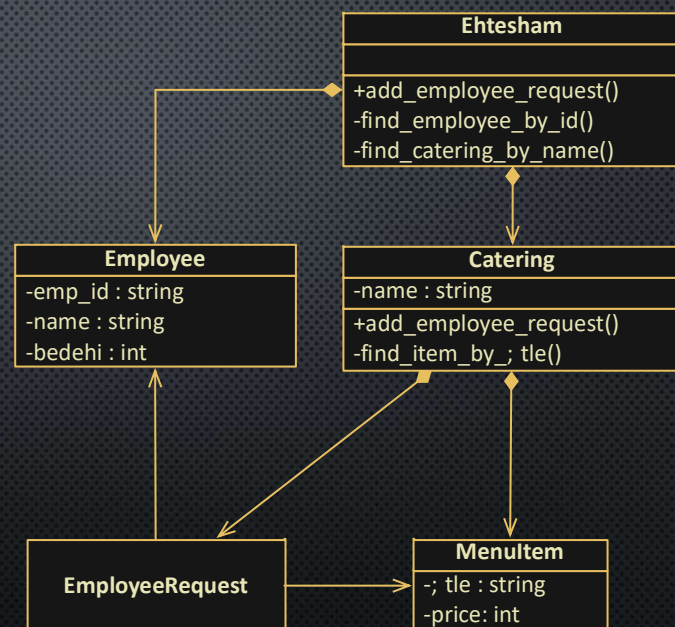
empid 1001

emp

cat

ورودی نمونه

request	1001	Farsi	Kabab
request	1002	Farsi	Kabab
request	1003	Alaa	Maahi
request	1002	Farsi	Jooj
catorder	Farsi		
catorder	Alaa		
delivery	Farsi	10000	
bedehi			
delivery	Alaa	10000	
bedehi			



ehtesham5.cpp

