**Exception**

Biz code yazarkən bilərək və ya bilməyərək müəyyən səhvlərə yol veririk. Bu səhvlər isə müəyyən bug’lara, error’lara gətirib çıxarır. Bu error’lar isə özlüyündə 2 yerə bölünür compile time və runtime olan error’lar. compile time olan error’lar rahatlıqla aradan qaldırıla bilir çünki zatən biz kod yazmağa davam edərkən compiler tərəfindən oxunan kodlar bizə hər hansısa problemlər varsa compile zamanı yerindəcə bildirir. Lakin Runtime olan errorlar yəni Exception’lar isə mütləq catch olunmalıdır. Əks təqdirdə program “partlayacaq”. Müəyyən şərt daxilində biz throw keyword vasitəsi ilə Exception’dan instance alınmış object’i istifadə olunacağı yerdə “partlamasına” səbəb ola bilərik.

if(true){

throw new Exception(“something is running wrong”);

}

Exception’un instance alınması üçün 3 overload hal var. Parametr qəbul etməyən(default,parameterless), string message qəbul edən və həm string message həm də innerException qəbul edən.

Exception exceptionObj = new();

Exception exceptionObj2 = new(“message”);

Exception exceptionObj3 = new (”message”, new Exception());

**Try & Catch**

Exception baş verə biləcək kod blokunu try daxilində yazıb, exception’u isə əgər baş verərsə catch vasitəsi ilə tutub onun üzərində gedişata uyğun bir iş görəcəyik. Try daxilindəki exception baş verdiyi sətrdə program avtomatik olaraq catch hissəyə düşərək oradaki kod blokunu işə salacaq və baş verən exception’ın instace’nı catch mötərizəsi daxilində tutulacaq .

try{

}

catch(Exception ex){

}

Burada exception object’i olan ex vasitəsi ilə biz ex.Message yazaraq exception mesajına, ex.InnerException yazaraq exception’ın daxilində göndərilən exception’a(nullable’dır) və s. müraciət edə bilərik.

**Multiple Catch**

Biz hər exception type’ı üçün fərqli catch əməliyyatı həyata keçirmək istəyiriksə, ardıcıllıq olaraq ən sonuncu derived class ən üstdə, ən başda duran base class isə ən sonda yazılmaqla bir neçə catch bloku yaradılmalıdır. Bir-birindən asılı olmayan exception’lar üçün hər hansısa bir sıralama məcbur deyil.

try{

}

catch(NullReferenceException ex){

}

catch(SystemException ex){

}

catch(Exception ex){

}

**Finally**

Finally keyword vasitəsi ilə biz exception baş versə də, baş verməsə də mütləq işə salınmalı olan kod blokunu təyin edirik.

try{

}

catch(Exception ex){

}

finally{

}