Обект – съвкупност от конкретни стойности. Примерно обект рожден ден – има ден, месец , годива – това са properties на обекта

DateTime birthday = new DateTime(1966, 11, 27) –обект от тип datetime

С оператора new ние създаваме обект от клас

Класа е нещо по-голямо, което дефинира функционалностите на обектите

Като правим променлива ние правим обект от този клас – string, integers

Класа има информация – Data и Actions – действия, които можем да правим – те реално са методите, с които сме работили досега

Примерно – клас студенти. Обект е един от тези студенти.

\*\*\*{0} - 0:d-MMM-yyyy – така може да принтираме DateTime

Когато печатаме обект, трябва да му кажем как да изглежда, така го „стрингосваме“

Substract – вади разлика между 2

\*\*\*TimeSpan – клас, разлика между 2 дати.

TimeSpan ageDiff = **peterBirthday.Subtract(mariaBirthday);**

Cw {0}, ageDiff.Days – изкарва дните разлика. В класа имаме и часовете, секундите и тн

DateTime.ParseExact - взема стринг и го превръща в DateTime обект. Добавяме CultureInfo.InvariantCulture – този тип дата няма значение къде в целия свят се намираме

За CultureInfo трябва да добавим класа System.Globalization

Date.DayOfWeek – вади деня на седмицата. Ключа означава, че имаме готова стойност, можем да вземем инфо. Куба значи че е метод, можем да направим действие с този клас. С ctrl + space се изкарват всички възможни действия, които могат да се направят след .

Math e клас, с методи, които можем да използваме.

System.Collections са данни като листове, речници и т.н.

System.Net – дава функционалност за свързване със сайтове, сваляне на файлове и тн.

DateTime today = DateTime.Now – връща точно сегашния момент. Статичен. Постоянно е там и постоянно брои кое точно е днес

Имаме статични и нестатични класове.

Статичните можем да ги извикаме веднага и те работят, без да ги дефинираме

Нестатичните трябва да напишем new за да можем да ги дефинираме

Random rnd = new Random(); - нестатичен

int randomNumber = rnd.Next(1, 100);

Random не знае какво е, получава стойност в момента, в който се стартира. То не знае какво е, докато не се стартира

Random – работи по странен начин чрез DateTime, компа предварително го измисля и точно в определен тик, определена част от секундата, е определено и реално не е точно random, а е предсказуемо, но се мени изключително бързо – по тикове. ВИНАГИ РАБОТИ С ЧИСЛО отдолу, но можем да го накараме отгоре да върне друга информация.

WebClient webClient = new WebClient();

webClient.DownloadFile("http://www.introprogramming.info/wp-content/uploads/2015/10/Intro-CSharp-Book-v2015.pdf", "book.pdf"); - създай този клиент, свали този файл, след това стартирай за да видиш дали работи. Проверява файла дали е там, дали правилно работи, реално е диагностика, дали успешно сме свалили файла.

WebClient

BigInteger – огромна целочислена стойност

Всяка програма, която правим е в нов клас – Program

Компилатора търси клас, в който стартира Main метод. Методите не могат да съществуват извън класове.

При наименоване трябва да ползваме долна черта, а не тире.

Нов клас се пише извън класа на програмата и после можем да го ползваме.

Class program

{

Point p = new Point(){ x= 5, y = 7}

Cw(p.X)

Cw(p.Y)

Cw(p) = cw(p.ToString()); - принтира типа данни - Point

}

Class Point

{

Public int x { get; set;}

Public int y { get; set;}

}

Person angel = new Person();

angel.FirstName = “Angel”;

angel.LastName = “Georgiev”;

При наименоване на properties на класа ползваме Pascal Case

Можем в класа да напишем метод и после клас.Метод извиква този метод

? : -тернарен оператор

Нестатичен клас със статичен метод не е ОК.

Ctrl +r +r – rename, за преименуване на променливите