**Batch 59**

**Java 01**

**19.02.2022**

**Genel Hatirlatmalar**

**Java Giris**

****

**Genel Hatirlatmalar**

**1. Derslere Hazirlanin ve Zamaninda Katılin**

**2. Dersi Dikkatli Dinleyin** 

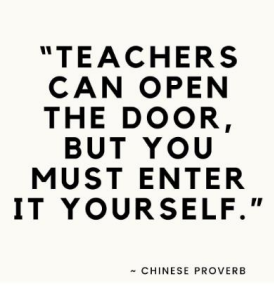
**3. Derste Aktif Olun**

**4. Anlamadiklarinizi Sorun**

**5. Ödevlerinizi Yapin (Kod yazma araba kullanma gibidir)**

**6. Her Dersten Sonra Tekrar Yapin**

**Genel Hatirlatmalar 7. Basari = Egitim + Calismak**

**8. Grup calismalari yapin, En iyi ogrenme yontemi ogretmektir** 

**9. Mentoring toplantilarini kacirmayin**

**10. Maillerinizi gunluk kontrol edin**

**11. Yoklama yapiliyor zooma isminizle girin 12. Teknik destek slack @technical support**

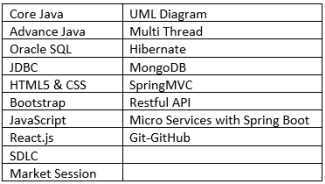
**13. Ders esnasinda canli destek**

**Free : Nur, Zafer , Y.Selim**

**Batch 59 : Elif, Merve, Feyza, Yusuf, Kenan**

**14. Customer service +1 917 768 74 66**

**Gorulecek Dersler Automation Engineer:**

**Java Developer** 

**Mobile Developer**

****

**Mentoring Mentoring toplantıları her hafta team tarafından ortak belirlenen gün ve saatte düzenli şekilde yapılmaktadır.**

✓ **Mentoring faaliyetleri STUDENT COACHING (öğrenci danışmanlığı) olarak yapılmaktadır.**✓ **Mentoring faaliyetlerinde...**

➢ **Haftanın görülen derslerin değerlendirmesi...**

➢ **Derslerle ilgili döküman desteğinin sağlanması....**

➢ **Ödev proje vs çalışmaların takip edilmesi...**

➢ **Team work’lerin takip edilmesi...**

➢ **FlipGrid çalışmalarının takip edilmesi...**

➢ **Java verbal çalışmalarının takip edilmesi...**

➢ **Java coding çalışmalarının takip edilmesi...**

➢ **Interview çalışmalarının takip edilmesi...**

**DÜZENLİ OLARAK YAPILMAKTADIR....**

**Ders Isleyisi - Bilmeniz Gerekenler**

**1. Maillerinizi gunluk kontrol edin**

**2. Dersleri zoom’dan izliyoruz ama mesajlasma icin slack kullaniyoruz - Iki slack kanalimiz var** 

**- Direk mesaj**

**- Kod paylasma (snippet)**

**- Mesaj silme ve edit**

**- Pin yapma**

**3. Google Clasroom** 

**- Tum ders notlari, zoom linki ve videolar Google**

**Classroom’dan paylasilacak**

**- Maillerinize davetiye gonderildi**

**- Youtube videolari**

**Ders Isleyisi - Bilmeniz Gerekenler 1-Ders esnasında öğrencilerin dikkatini dagitacak paylasimlar yapmayin** 

**2-Ders esnasında ders ve konu dışında paylasım yapmayınız.**

**3-Diyaloglarınızda asgari nezaket ve saygı kurallarına azami**

**dikkat ediniz.** 

**4-Ders esnasında ders hocasına direct mesaj yazmayınız**

**5-Derste code paylaşırken SNIPPET ve screenshot kullanmaya**

**dikkat ediniz**

**6-Dersi iyi takip ediniz, öncesinden sorulmuş ve cevaplanmış soruyu**

**tekrar sormamaya azami gayret gösteriniz.**

**7-CODE ve SYNTAX hatalarınız için MENTOR'lerimiz, KURULUM**

**hatalarınız için TECHNICAL SUPPORT yardımcı olacaktır.**

**8-CODE ve SYNTAX hatalarınızı DM olarak değil benzer hataları**

**alanların da yararlanması için öğrenci yardımlaşma channel'den**

**paylaşınız.**

**Ders Isleyisi - Bilmeniz Gerekenler**

**1. Ders tam zamaninda baslar.** 

**2. Dersin basinda 10 dakika bir**

**onceki gunun kisa tekrari yapilir**

**3. Her konu bittiginde ertesi gun**

**kisa tekrardan sonra**

**Socrative testi yapilir ( 10 -15**

**dk) sonra o sorular cozulerek**

**konu tekrari yapilir**

**Programlama Dili Nedir ?** 

**Hareket**

**Codes JDK Binary Islem (Islem Yap)** Compile

package rentalstore; 

class Customer {

private String \_name;

}

public String getMovie(Movie movie) {

Movie.*NEW\_RELEASE),*

return movie.getTitle();

(Derleme)

11010101010101001010001010 10101001011110000010001001 01101010101010010111000110 01000101001010111000100000 01010010101110001001010111

**Nicin Java ?**

**1- Ogrenmesi kolay**

**2- Dunyada en cok kullanilan programlama dili Sun'a göre 3 milyar cihaz Java kullanıyor. Şu anda Java'nın kullanıldığı birçok cihaz var.** 

**Bunlardan bazıları şu şekildedir:**

**- Acrobat reader, medya oynatıcı, antivirüs vb. - Masaüstü Uygulamaları**

**- Bankacılık uygulamaları gibi Kurumsal Uygulamalar**

**- Cep Telefonu**

**- Akıllı kart uygulamalari**

**- Robotik uygulamalari**

**- Oyunlar**

**3- Java “Object Oriented Programming (OOP)” Language’ dir.**

**Object Oriented Programming Nedir? Objects (Nesne) Application (Urun)1- Feature (Fields veya Variables)**

**Pasif ozellik (renk,sekil,isim)**

**2- Functionality (Method)**

**Aktif ozellik (tasima,degistirme)**

**Bir Object Nasil Olusturulur?**

****

**Birden fazla Obje birlestirilir**

**Class(Object Kalibi) Object**

**Field Method** 

**(Variables) (Functions)**

**Application**

**Object Nasil Kullanilir ?**

****

**Ogretmen**

**Dersler Personel** 

**Ogrenci Notlar**

**Bir Class Hangi Bolumlerden Olusur? 1 – Class Declaration**

**2**

**1**

**2 – Curly braces : Suslu parantez**

**3 – Class Body : Suslu parantezler**

**arasinda kalan ve kodlarimizi** 

**3**

**yazdigimiz bolum**

**2**

**Bir Class’in Icinde Neler Bulunur? 11 – Field / Variables 2 2 – Main Method** 

**3 3 – Method**

**Class Olustururken (Declaration) Kullanilan Keyword’ler**

**public class MyFirstClass { }**

**1 2 3 4**

**1 public : Access Modifier (Erisim duzenleyici) : class’a kimlerin erisebilecegini belirler. Public olursa her yerden erisilebilir**

**default : Sadece bulundugu Package’den kullanilabilir**

**2 class : Yazdigimiz kodun class oldugunu belirtir**

**3 MyFirstClass : Olusturdugumuz class’in ismidir. Class’a istedigimiz ismi verebiliriz ancak isim verilirken genelde class’da yapilan isleme uygun bir isim secilmesine dikkat edilir.**

**Isim mutlaka buyuk harfle baslar, birden fazla kelimeden olusursa sonraki kelimelerin ilk harfleri de buyuk harf yazilir (Camel Case)**

**4 Body (Class Body) : { } arasinda kalan kodlarimizi yazdigimiz bolumdur**

**Method Olusutururken Kullanilan Keyword’ler public int myFirstMethod () { }**

**1 2 3 4 5**

**1 public : Access Modifier (Erisim duzenleyici):methoda’a kimlerin erisebilecegini belirler private: Sadece bulundugu class’da kullanilabilir**

**protected : Sedece icinde bulundugu class ve child class’lardan kullanilir 2 Int : Return Type, methodun ne urettigini ve bize dondurdugunu belirtir**

**3 myFirstMethod :Olusturdugumuz method’un ismidir. Isim mutlaka kucuk harfle baslar, birden fazla kelimeden olusursa sonraki kelimelerin ilk harfleri buyuk harf yazilir (Camel Case)**

**4 () parantez: Methodlarda isimden sonra parantez kullanilir ve gerektiginde parantez icinde parametre yazilir.**

**5 Body (Method Body) : { } arasinda kalan kodlarimizi yazdigimiz bolumdur**

**Main Method public static void main(String[ ] args) { }**

➢ **main method, java’nin calismaya basladigi** 

**giristir. (Entry Point)**

➢ **main method olusturulurken yazilmasi gereken**

**syntax (kod dizimi) degistirilemez**

➢ **Parantez icinde yazilan (String[] args) java’nin**

**calismasi icin gerekli olan parametreleri**

**barindirir ve olmasi sarttir.**

**Araba Motor**

**Java Project Main Method**

**Yorum Cumlesi (Comment) Nasil Eklenir ?**

➢ **Comments : Java tarafindan** 

**calistirilmayan, amaci kodlarin**

**aciklanmasi veya bir konuda**

**bilgi vermek olan cumlelerdir**

➢ **Genelde iki kullanim vardir**

**1) Tek satirlik comment**

**2) Cok satirlik comment**

**Data Nedir?**

**Data bilgisayar tarafından işlenen (processed) veya depolanan (stored) bilgidir. **

**Java’nin kullandığı (use) veya ürettiği (produce) her şey data’dır. **

**Bit bit hafizadaki en küçük data parçasıdır. Her “bit” bir binary value içerir, 0 veya 1. Note: 8 bit =1 byte**

**Bit** 

**Byte**

**Memory**

**(Hafiza)**

Batch 60

Java 02

19.02.2022 **Java Giris Variables**

**IntelliJ Kullanim 1- Proje olusturma**

**File -- New -- Project -- (Java Project) Next -- java2022WinterTr -- finish**

**2- Package (paket) olusturma**

**src dosyasina sag click -- New -- Package -- day01variables -- finish**

**3- Class olusturma**

**day01variables dosyasina sag click -- New -- Class -- C01\_Variables01 -- finish 4- Main method olusturma**

**public static void main(String[] args) yazarak main methodu olusturalim**

**Variables (Degisken) Olusturma Declaration**

**Variable bellekte (memory) ayrilmis olan alanin (reserved area) adidir.**

**Variable icinde deger saklayan bir konteynirdir (container).** 

**Bir değişkende saklanan değer, program yürütülürken değiştirilebilir.**

**Java'da, tüm değişkenler kullanılmadan önce deklare**

**edilmelidir (variable declaration)**

**Variable declaration icin iki seyi belirtmemiz gerekiyor** 

**1- Data type (data turu)**

**2- Variable Name (degisken ismi)**

**Variables Deger Atama (Assignment)**

**Varolan bir variable’a deger atamaya assignment (atama) denir.**

**1- Deger atamasi yapilirken data turune uygun deger atanmalidir. Diger turlu Java hata verir. **

**Variables Deger Atama (Assignment)**

**2- Ilk once declaration, daha sonra atama yapilabilir.**

**3- Bir defa declaration yapildiktan sonra, birden fazla** 

**atama yapilabilir. Java son degeri tutar, oncekini siler.**

****

**Variables Deger Atama (Assignment)**

**4- Ayni data turunde birden fazla variable tek komutla deklare edilebilir. **

**5- Ayni data turunde birden fazla variable tek komutla deklare edilip deger atanabilir.**

**Data Types**

**Java’da iki data tipi kullanilmaktadir**

**- Primitive Data Types : boolean, char, byte, short, int, long, float ve double- Non- Primitive Data Types : String,**

**ilerleyen derslerde gorecegimiz primitive olmayan Array, List, Object gibi her data non-primitive’dir.**

**Primitive Data Types**

**1) boolean Data Type: true veya false barindirir. Hafizada 1 bit kullanir Sadece dogru veya yanlis seklinde cevap verilebilecek variable’larda kullanilir boolean isExpensive = true;**

**boolean isCold = false;**

**2) char Data Type : Tek karakter barindirir. Hafizada 16 bit kullanir Harf, sayi veya sembol bakilmaksizin sadece 1 karakter kullanacak variable’larda kullanilir**

**char letter = ‘a’;**

**char digit = ‘3’;**

**char cymbol = ‘#’;**

**Note: char degerlerini single quote arasina yazilir.**

**Primitive Data Types**

**3) byte Data Type: -128 den 127’e (dahil) tamsayilar icin kullanilabilir. Hafizada 8 bit kullanir**

**byte age = 73;**

**4) short Data Type: -32.768 den 32.767’e (dahil) tamsayilar icin kullanilabilir. Hafizada 16 bit kullanir**

**short koyNufusu = 27,324;**

**5) int Data Type: -2.147.483.648 den 2.147.483.647’e (dahil) tamsayilar icin kullanilabilir. Hafizada 32 bit kullanir**

**int turkiyeNufusu = 67,324.564;**

**6) long Data Type: -9,223,372,036,854,755,808 den ,223,372,036,854,755,807’e (dahil) tamsayilar icin kullanilabilir. Hafizada 64 bit kullanir**

**Primitive Data Types**

**7) float Data Type: Kucuk ondalik sayilar icin kullanilabilir. Hafizada 64 bit kullanir float floatVar2 = -2.123456f;**

**Not: float sayilarin sonunda “ f ” yazilmalidir, yazilmazsa java sayiyi double kabul eder**

**8) double Data Type: Buyuk ondalik sayilar icin kullanilabilir. Hafizada 64 bit kullanir double doubleVar2 = -2.12345679078000000000123**

**Non-Primitive Data Type**

**String Data Type:**

**String pes pese dizilmis char’lardan olusur. Kelimeler, cumleler, matematiksel islem yapilmayacak sayisal degerler de String olarak tanimlanabilir**

**String okulAdi = “Yildiz Koleji, Cankaya Ankara #”;** 

**String telNo = “5321234567”;**

**String ilkHarf = “A”;**

**Note: String’ler cift tirnak (double quotes) arasina yazilir.**

**Note: Baska non-primitive data type’lar da var, daha sonra ogrenecegiz.**

**Primitive VS Non-Primitive Data Types 1) Primitive’ler sadece value icerir, non-primitive’ler value ve methodlar icerir. 2) Primitive’ler kucuk harf ile, non-primitive’ler buyuk harf ile baslar.**

**3) Primitive’leri Java olusturur biz primitive data turu olusturamayiz.**

**Non-primitive’leri biz de olusturabiliriz, Java da olusturabilir. Or: String’i Java olusturmustur.**

**4) Primitive’lerin buyuklukleri data type’ing gore sabittir. non-primitive’ler icin sabit buyukluk soz konusu degildir.**

Batch 60

Java 03

19.02.2022

**Kullanicidan Deger Alma Data Casting**

**Increment/ Decrement**

**Önceki Dersten Aklımızda Kalanlar**

**1. OOP concept : nesne tabanli programlama demektir, biz olusturdugumuz class’lar sayesinde objeler uretebiliriz ve bu objeler I birlestirerek kompleks uygulamalar gelistirebiliriz(lego gibi)**

**2. Clas hangi bolumlerden olusur ?**

**-class declaration : keyword ler sayesinde class’I kimlerin kullanabilecegini gorebiliriz, class ismi buyuk harfle baslar ve CamelCase seklinde yazilir**

**- { } curly braces / suslu parantez : Classin nerede baslayip bitttigini gosterir - Class Body : curly braces arasinda kalan ve kodlarimizi yazdigimiz bolumdur 3- Class icerisnde neler olur ?**

**- Main method :arabanin motoru gibidir, Java kodlarimiz calistirmaya main method’dan baslar**

**- normal method’lar : method’lar bizim adimiza belirledigimiz islemleri yapip, islem sonunda da istedigimiz sonucu bize dondururler**

**- variable : bizim icin degerleri saklayan konteynir’lardir**

**Önceki Dersten Aklımızda Kalanlar**

**4- variable : bizim istedigimiz deger koyabilmemiz icin hafiza da ayrilan bolumun adidir. Ornegin bir oyunda level bilgisi icin bir variable tanimlarsak, oyunun hangi asamasinda olursa olsun level variable’ina baktigimizda icinde level degerimizi gorebiliriz Biz programimiz icerisinde ne zaman variable ismini yazsak, java o variable yerine en son atanan degeri kullanir.**

**5- Data turleri**

**- primitive data turleri : boolean, char, byte, short, int, long, float, double - non-primitive data turleri : String (ilerde pek cok cesidini gorecegiz, biz de istersek non-primitive data turu olusturabiliriz)**

**6- iki data turu arasindaki farklar**

**- p’ler sadece depolama yapar, np’ler ise hem depolama yapar hem de kendilerine ozel methodlar sayesinde istedigimiz degisimleri yaparlar**

**- p data turlerinin isimleri kucuk harfle baslar, np data turlerinin isimleri buyuk harfle baslar**

**- p’ler 8 tanedir ve biz yeni p data turu uretemeyiz, ancak np’leri java da uretebilir biz de uretebiliriz, dolayisiyla np data turu sayisi sinirlandirilamaz**

**Variable ve Method’lar Nasil Adlandirilir**

**1. Java variable isimleri case sensitive (Buyuk kucuk harfe duyarlidir)dir. “money”, “Money” veya “MONEY” birbirinden farklidir**

**2. Java variable isimleri “harf”,“$” veya “\_” ile baslamalidir.**

**Fakat “$” ve “\_” ile baslamak tavsiye edilmez.**

**3. Java variable isimlerinde, ilk harften sonra sayi, “$” ve “\_” kullanilabilir. 4. Variable isimleri icin Java’ya ozel terimler (key word) kullanilamaz. (int, for, if, import vb). 5. Variable isimleri kucuk harflerle baslar, camel case kullanilir**

**6. Variable isimleri 1’den fazla kelime iceriyorsa, ilk kelimeden sonraki her kelimenin ilk hafi buyuk harf ile baslamalidir. firstName, bigApple, ageJohnWalker gibi. Buna camelCase denir.**

**Memory (Hafiza) Kullanimi Javada kullanilan iki hafiza vardir** 

**Stack => small**

**Heap => huge**

**1- Stack Memory : primitive data tiplerine ait degerleri ve Non-primitive datalara (Object) ait referanslari(adres) barindirir**

**2- Heap Memory : Non-primitive data’lari depolamak(store) icin kullanilir**

**Memory (Hafiza) Kullanimi**

**reference of an Object** 

**String long**

**int**

**Object byte Array boolean**

**Stack Heap**

**Variables Class Work**

**1- Farkli 3 data turunde variable olusturun ve bunlari yazdirin 2- isim ve soyisim icin iki variable olusturun ve bunlari**

**isminiz : Mehmet**

**soyisminiz : Bulutluoz**

**seklinde yazdirin**

**3- Iki farkli tamsayi data turunde 2 variable olusturun bunlarin toplamini yazdirin 4- Bir tamsayi ve bir ondalikli variable olusturun ve bunlarin toplamini yazdirin 5 – char data turunde bir variable olusturun ve yazdirin**

**6- Bir tamsayi, bir de char degisken olusturun ve bunlarin toplamini yazdirin.**

**ASCII Table**

**Variables Class Work**

**Interview Question**

**1- Verilen sayi1 ve sayi2 variable’larinin degerlerini degistiren (SWAP) bir program yaziniz**

**Orn : sayi1=10 ve sayi2=20;**

**kod calistiktan sonra**

**sayi1=20 ve sayi2=10**

**2- Verilen sayi1 ve sayi2 variable’larinin degerlerini 3.bir variable olmadan degistiren (SWAP) bir program yapiniz**

**Kullanicidan Deger Alma 1) Scanner scan = new Scanner( System.in );**

**scan : olusturdugumuz scanner’in ismidir ve istedigimiz ismi vermemiz mumkundur. Ancak genelde scan ismi kullanilir.**

**Bu tur isimlendirmelerde genel kurallara uymamiz kodumuzun anlasilabilir olmasi acisindan faydali olacaktir.**

**2) System.out.println( “Lutfen 100’den kucuk pozitif iki tamsayi giriniz” ); Kullaniciya girmesini istedigimiz degerler icin aciklayici bilgi vermeliyiz.**

**Burada aciklama olarak ne yazdirsak kodumuz calisir, hatta birsey yazdirmasak da calisir ancak kullanici kendisinden ne istedigimizi bilmezse deger girmesi gerektigini veya ne tur bilgi girmesi gerektigini bilemez**

**Kullanicidan Deger Alma**

**3) scan.nextInt() ile girilen degerleri alabiliriz. Istedigimiz data tipine gore next’ten sonra yazilacak kisim degisir.**

**int num1 = scan.nextInt()**

**int num2 = scan.nextInt()**

****

**Kullanicidan Deger Alma Sorular**

**Soru 1) Kullanicidan iki tamsayi alip bu sayilarin toplam,fark ve carpimlarini yazdirin Soru 2) Kullanicidan karenin bir kenar uzunlugunu alin ve karenin cevresini ve alanini hesaplayip yazdirin**

**Soru 3) Kullanicidan yaricap isteyip cemberin cevresini ve dairenin alanini hesaplayip yazdirin Soru 4) Kullanicidan dikdortgenler prizmasinin uzun, kisa kenarlarini ve yuksekligini isteyip prizmanin hacmini hesaplayip yazdirin**

**Soru 5) Kullanicidan ismini ve soyismini isteyip asagidaki sekilde yazdirin Isminiz : Mehmet**

**Soyisminiz : Bulut**

**Kursumuza katiliminiz alinmistir,tesekkur ederiz**

**Soru 6) Kullanicidan ismini ve soyismini alip aralarinda bir bosluk olusturarak asagidaki sekilde yazdirin**

**Isim – soyisim : Mehmet Bulutluoz**

**Soru 7) Kullanicidan ismini alip isminin bas harfini yazdirin.**

Batch 59-60

Java 04

23.02.2022

**Data Casting**

**Increment / Decrement Matematiksel Operatorler**

**Önceki Dersten Aklımızda Kalanlar**

**1. Scanner class’I kullanicidan bilgi alamk icin kullanilir.**

**2. Uc adimda kullanicidan degeri aliriz**

**- Scanner olusturma, parameter olarak System.in yazmaliyiz**

**- Kullaniciya ne girecegini soyleyen bir mesaj yazdirma**

**- Olusturdugumuz scan objesi ve girilecek dataya uygun next method’u ile kullanicinin girdigi degeri alip, gelen dataya uygun data turunde olusturdugumuz variable’a atama yapariz**

**3- Kullanicidan aldigimiz deger metin ise next( ) veya nextLine( ) method’u kullanilir. Next method’u sadece ilk bosluga kadar olan metni alirken, nextLine tum satiri alir 4- Java da kullanilan 2 tur hafiza vardir.**

**stack- primitive dataturundeki variable’larin aldigi degerler ve non-primitive’lerin referanslari bulunur**

**Heap : Non primitive data turundeki datalar**

**Data Casting / Veri Sinifi Degistirme**

➢ **Java’da kod yazarken bir veri tipinden diğer bir veri tipine aktarım yapmamız gerekebilir.**

➢ **Veri tiplerinde bir variable’a , olusturuldugu data tipinden farkli bir data turunden deger atanmasina Data Casting denir.**

➢ **Data casting yaparken aklimizdan cikarmamamiz gereken konu data tiplerinin sinirlaridir. Data tipinin sinirlarini asan data casting islemlerinde hata almamamiz icin dikkat etmemiz gereken bazi durumlar olacaktir.**

➢ **Hatirlayacagimiz sekilde Java’da sayilarla ilgili data tiplerinin siralamasi su sekildeydi byte < short < int < long < float(ondalıklı) < double(ondalıklı)**

**Data Casting / Veri Sinifi Degistirme 1) Auto Widening (Otomatik Genisletme)**

**Dar veri tipinden daha genis bir very tipine**

**gecmek iztedigimizde Java donusumu**

**otomatik olarak yapacaktir.**

**byte short int long float(ondalıklı) double(ondalıklı) Orn : byte num1 = 12;**

**short num2 = num1; // yazdirirsak 12 olarak yazdirir**

**int num3 = num2; // yazdirirsak 12 olarak yazdirir**

**float num4=num3; // yazdirirsak 12.0 olarak yazdirir**

**double num5=num4; // yazdirirsak 12.0 olarak yazdirir**

**Data Casting 2) Explicit Narrowing (Manuel Daraltma)**

****

➢ **Genis veri tipinden daha dar bir veri tipine gecmek istedigimizde Java donusumu otomatik olarak YAPMAYACAKTIR.**

➢ **Bu durumda Java Casting’in bir problem olusturabilecegini varsayarak sizden MANUEL ONAY isteyecektir.**

➢ **Narrowing Casting bazi datalari kaybetmemize yol acabilir, bazen de sayiyi kendi sinirlari icinde kalan baska bir sayiya donusturebilir**

**Data Casting**

**Soru 1 ) byte veri tipinde bir degisken olusturun, short,int,float ve double data tiplerinde birer degisken olusturup adim adim widening yapin ve yazdirin**

**Soru 2 ) int veri turunde bir degisken olusturun ve adim adim narrowing yapin ve yazdirin Soru 3 ) Float data turunde bir variable olusturun ve yazdirin**

**Soru 4 ) double 255.36 sayisini int’a ve sonra da olusturdugunuz int sayiyi byte’a cevirip yazdirin**

**Soru 5 ) int 2 sayiyi birbirine boldurun ve sonucu yazdirin**

**Soru 6 ) int bir sayiyi double bir sayiya bolun ve sonucu yazdirin**

**Soru 7 ) Farkli data tipleri ile islem yapip, sonuclarini yazdiralim**

**Increment / Bir Variable’in Degerini Artirma Yontemleri veya ?**

**?**

**veya**

**? ?**

**Decrement / Bir Variable’in Degerini Azaltma Yontemleri veya ?**

**?**

**veya**

**? ?**

Batch 59-60

Java 05

24.02.2022

**Pre & Post Increment Matematiksel Operatorler Modulus**

**Önceki Dersten Aklımızda Kalanlar** 

**1. Data Casting (Veri turunu degistirme): Java her data turunu birbirine cevirmez, ornegin Boolean bir degiskene String bir deger atayamayiz.**

**Ancak sayisal veri turlerini birbirlerine cevirebiliriz**

**- Eger variable turu(esitligin solu) daha kapsamli ise, Java bu casting islemini otomatik olarak yapar (Auto Widening)**

**- Eger deger’in turu (esitligin sagi) daha kapsamli ise Java bunu otomatik olarak yapmaz. Cunku daha kapsamli bir data turunden daha dar kapsamli bir data turune gecis sirasinda data kayiplari veya farkli deger alma ihtimali olusur.**

**Bu durumda Java sorumlulugu manuel olarak almamizi ister. Sorumlulugu alabilmek Icin esitligin sagina parantez icerisinde variable’in data turunu yazariz**

**double sayi1=10.28;**

**int sayi2 = (int)sayi1 ;**

**2- Increment ve Decrement : artirma veya azaltma demektir**

**- sayi1 = sayi1+3; bu cok tercih edilmez**

**- sayi1 += 3; genelde bu tercih edilir**

**- sayi1++; sadece 1 artirip azaltacaksak bunu kullaniriz**

**Pre-Increment & Post Increment** ➢ **Pre-Increment ve Post Increment operatorlerinin her ikisi de artirma islemi icin kullanilir** ➢ **Pre-Increment isleminde variable statement’da kullanilmadan once artirilir veya azaltilir**

**Output : 16**

➢ **Post Increment isleminde variable statement’da kullanilir, sonra artirilir veya azaltilir Output : 15**

**Javada Matematiksel Operatorler**

**1- Ustel islemler**

**2- Parantez ici**

**3- Carpma-Bolme**

**4- Toplama-cikarma**

**Ornek 1 : Ornek 2 :**

**38 / 2 \* ( 4 + 3 ) \* 2 =**

**8 + 2 \* ( 14 – 6 / 2 ) - 12 =**

**Modulus % Modulus islemi bir bolme isleminde kalan sayiyi bize verir**

****

****

**Modulus % Soru ) Kullanicidan 4 basamakli bir sayi alin ve rakamlar toplamini bulup yazdirin**

Ipucu 1:

Sayi % 10 => Bize son basamagi verir

**538 % 10 = 8**

Ipucu 2:

Int Sayi /10 => Bize son basamak haric sayiyi verir

**int** sayi=538;

**sayi = sayi / 10** =>

**sayi’**ya 53 degerini atar

**Wrapper Class Java primitive data turleri ile methodlari kullanabilmemiz icin Wrapper class’lari olusturmustur.**

**Character,Byte,Integer,Short,Float,Double primitive data turleri icin olusturulan wrapper class’lardir.**

****

Batch 59-60

Java 06

25.02.2022

**Concatenation**

**Relational Operators Conditional Operators**

**Önceki Dersten Aklımızda Kalanlar** 

**1. Wrapper Class : primitive data turlerine ait method yoktur. Java primitive data turundeki variable’larin Wrapper Class’larine yaparak o turler icin de hazir bazi method’lar olusturmustur.**

**2. Pre Increment- Post Increment : bu ikisi de ++ , veya – icin gecerlidir. Java yukaridan asagiya, soldan saga dogru gider. Eger bir satirda birden fazla islem varsa, once hangisini yapacagini bilmesi gerekir**

**int sayi2 = sayi1++; once sayi1’in eski degerini sayi2’ye atar, sonra sayi1’i 1 artirir sout(++sayi1); once sayi1’in degerini 1 artirip, sonra yeni degerini yazdirir 3- Modulus % : bir bolme isleminde kalan’I verir (matematikdeki mod islemi gibidir) Bizim en cok kullandigimiz yerler**

**- bir sayinin tek mi cift mi oldugunu bulmak**

**- bir sayinin verilen bir sayiya tam bolunup bolunemedigine bakmaz**

**- %10 yaparak sayinin birler basamagini almak**

**4- bir sayinin rakamlari toplamini bulmak icin icin, basamak sayisi miktarinca su islemeler tekrar edilir**

**- %10 ile birler basamagini bulmak**

**- bulunan basamaktaki rakami rakamlar toplamina eklemek**

**- toplama ekledigimiz rakamdan kurtulmak icin sayi/10 yapmak**

**Socrative Quiz**

**1) https://b.socrative.com/login/student/ adresine gidin 2) Room Name BULUTLUOZ yazin**

**3) Isminizi yazin**

**4) Done butonuna basin**

**Sure : 15 Dakika**

**Concatenation / (String Datalari Birlestirme)**

**Birden cok String’i + isareti ile topladiginizda Java bu String degiskenleri birlestirerek yeni bir String olusturur**

****

****

**Not : Eger matematiksel bir islemin icinde String kullanilirsa, matematikteki oncelikler dikkate alinarak islem yapilir. Sira String ile toplamaya geldiginde toplama yerine Concatenation uygulanir**

**Hello**

**23**

**5**

**Hello**

**Hello**

**5**

**Hello**

**6**

**Concatenation**

**Soru 1 ) Asagida verilen variable’lari kullanarak istenen sonuclari yazdiran programlari yaziniz.**

**Variables**

**String str1= “Java”; String str2= “Guzel”; int sayi1=5;**

**int sayi2=4;**

**Istenen Yazilar**

**1) Java Guzel 54 2) Java 5 Guzel**

**3) Java 94**

**4) Java 19**

**5) 54 Guzel**

**Relational Operators /(Karsilastirma Operatorleri) = Assignment (Atama yapar) operatoru**

**int num1=3; num1 degiskenine 3 degerini atar String str1 = “Ali” + “ ” + “Can”; str1’e Ali Can degeri atar**

**c = c+5; c’nin degerini 5 artirir ve son degeri c’ye atar**

**== Cift esittir isareti / karsilastirma (Comperison) operatoru**

**boolean sonuc1 = 5+2 == 7; sonuc1 degeri true olur boolean sonuc2 = 5\*2 == 15; sonuc2 degeri false olur**

**Relational Operators /(Karsilastirma Operatorleri) != Esit degildir isareti**

**boolean sonuc1= 5+2 != 7; sonuc1 degeri false olur**

**System.out.println(5\*2 != 15); true yazdirir**

**> Buyuktur , >= Buyuk veya esittir**

**boolean sonuc1= 5+2 >= 7; sonuc1 degeri true olur**

**System.out.println(5\*2 > 15); false yazdirir**

**< Kucuktur , <= Kucuk veya esittir**

**boolean sonuc1= 5+2 < 7; sonuc1 degeri false olur**

**System.out.println(5\*2 < 15); true yazdirir**

**Conditional Operators / (Sart Operatorleri) && AND (ve) isareti**

**&& isareti ile birlestirilen tum ifadeler dogru ise sonuc true olur.**

**Diger tum durumlarda false doner. ( && operatoru mukemmeliyetcidir )**

**boolean sonuc1= (5+2 == 7) && (4+3 !=5) ; sonuc1 degeri true olur System.out.println((5\*2 != 15) && (5>7)); false yazdirir**

**|| OR (veya) isareti**

**|| isareti ile birlestirilen tum ifadeler yanlis ise sonuc false olur.**

**Diger tum durumlarda truee doner. ( || operatoru iyimserdir )**

**boolean sonuc1= (5+2 == 7) || (4+3 !=5) ; sonuc1 degeri true olur System.out.println((5\*2 == 15) || (5>7)); false yazdirir**

**& Ile && Arasindaki Fark & isareti kullanildiginda Java isaretin iki yanindaki mantiksal ifadelerin ikisini de kontrol eder. Bu islem kodumuzu yavaslatir**

**40<30 & 50==50 & 60>50**

**ilk karsilastirma yanlis olmasina ragmen Java tum karsilastirmalari kontrol etmeye devam eder.**

**&& isareti kullanildiginda ise Java en bastan kontrol etmeye baslar, mantiksal ifadelerin birinde yanlisi bulursa sonrakileri kontrol etme ihtiyaci duymaz. Bu islem kodumuzu hizlandirir**

**40<30 && 50==50 && 60>50**

**ilk karsilastirma yanlis oldugunu gorunce Java diger karsilastirmalari kontrol etmeden alt satira gecer.**

Batch 59-60 Java 07

26.02.2022 **If Statements**

**If Statements / (If cumleleri) Eger hava guzel olursa piknige gidecegiz. (guzel olmazsa karar yok)**

**Eger (hava guzel olursa) {piknige gideriz}her durumda alt satira gecer**

**If (boolean sart) {sart saglanirsa istenen kod} her durumda alt satira gecer**

**If Statements / (If cumleleri)**

**Not : If statement birden fazla olursa hepsi birbirinden bagimsiz olur. If cumlelerini birbirine baglamayi da ogrenecegiz.**

**Eger hava guzel olursa piknige gidecegiz. (guzel olmazsa karar yok) Eger Ali ararsa ona kizacagim. (aramazsa karar yok) Eger aksam mac varsa onu izleriz. (mac yoksa karar yok)**

****

**If Statements / (If cumleleri) Soru 1) Kullanicidan bir tamsayi isteyin ve sayinin tek veya cift oldugunu yazdirin**

**Soru 2) Kullanicidan gun isimlerinden birinin ilk harfini isteyin ve o harfle baslayan gun isimlerini yazdirin**

**Ornek: ilkHarf=P output = “Pazar, Pazartesi veya Persembe”**

**ilkHarf=S output = “Sali”**

**\*\*\* Buyuk kucuk harf problem olmamasi icin toUpperCase methodunu kullanin Soru 3) Kullanicidan gun ismini alin ve haftaici veya hafta sonu oldugunu yazdirin Ornek: gun=Pazar output = “Hafta sonu”**

**gun=Sali output = “Hafta ici”**

**\*\*\* String icin equals method’unu kullanin**

**Soru 4) Kullanicidan dikdortgenin kenar uzunluklarini isteyin ve dikdortgenin kare olup olmadigini yazdirin**

**Soru 5) Kullanicidan bir gun alin eger gun “Cuma” ise ekrana “Muslumanlar icin kutsal gun” yazdirin. “Cumartesi” ise ekrana “Yahudiler icin kutsal gun” yazdirin. “Pazar” ise ekrana “Hiristiyanlar icin kutsal gun” yazdirin**

**If Else Statements Eger hava guzel olursa piknige gideriz, yoksa evde otururuz.**

**Eger (hava guzel olursa) {piknige gideriz} yoksa {evde otururuz}**

**If (boolean sart) {sart saglanirsa istenen kod} else {sart saglanmazsa istenen kod}**

**If Else Statements Sorular**

**Soru 1) Kullanicidan dikdortgenin kenar uzunluklarini isteyin ve dikdortgenin kare olup olmadigini yazdirin**

**Soru 2) Kullanicidan bir karakter girmesini isteyin ve girilen karakterin harf olup olmadigini yazdirin**

**Soru 3) Kullaniciya yasini sorun, eger yas 65’den kucuk ise “emekli olamazsin, calismalisin”, 65’e esit veya buyukse “Emekli olabilirsin” yazdirin**

**Soru 4) Kullanicidan bir ucgenin uc kenar uzunlugunu alin eger uc kenar uzunlugu birbirine esit ise ekrana “Eskenar ucgen” yazdirin. Diger durumlarda ekrana “Eskenar degil” yazdirin.**

**If Else If … Statements**

**Eger soruyu biliyorsa Ali soruyu cozsun , o bilmiyorsa Veli biliyorsa Veli cozsun, o da bilmiyorsa Ayse biliyorsa, Ayse cozsun, o da bilmiyorsa Fatma biliyorsa, Fatma cozsun, o da bilmiyorsa kim isterse o cozsun.**

**Eger soruyu biliyorsa Ali soruyu cozsun , o bilmiyorsa Veli biliyorsa Veli cozsun, o da bilmiyorsa Ayse biliyorsa, Ayse cozsun, o da bilmiyorsa Fatma biliyorsa, Fatma cozsun, o da bilmiyorsa kim isterse o cozsun.**

**If (sart) {sart saglanirsa istenen kod} else if {sart saglanmazsa istenen kod} else if {sart saglanmazsa istenen kod} else if ( kac tane durum varsa else if ….. ) else {sart saglanmazsa istenen kod}**

**If Else If … Statements**

**Soru 5) Kullanicidan gun ismini yazmasini isteyin. Girilen isim gecerli bir gun ise gun isminin 1.,2. ve 3.harflerini ilk harf buyuk diger ikisi kucuk olarak yazdirin, gun ismi gecerli degilse “Gecerli gun ismi giriniz” yazdirin**

**Soru 6) Kullanicidan iki sayi isteyin, sayilarin ikisi de pozitif ise sayilarin toplamini yazdirin, sayilarin ikisi de negative ise sayilarin carpimini yazdirin, sayilarin ikisi farkli isaretlere sahipse “farkli isaretlerde sayilarla islem yapamazsin” yazdirin, sayilardan sifira esit olan varsa “sifir carpmaya gore yutan elemandir” yazdirin.**

**Soru 7) Kullanicidan 100 uzerinden notunu isteyin. Not’u harf sistemine cevirip yazdirin. 50’den kucukse “D”, 50-60 arasi “C”, 60-80 arasi “B”, 80’nin uzerinde ise “A”**

**Soru 8) Kullanicidan maas icin bir teklif isteyin ve asagidaki degerlere gore cevap azdirin. Teklif 80.000’in uzerinde ise “Kabul ediyorum” ,**

**60 – 80.000 arasinda ise “Konusabiliriz”,**

**60.000’nin altinda ise “Maalesef Kabul edemem” yazdirin**

**Nested If Else Statements**

**Eger calisan kadinsa 60 yasindan buyuk oldugunda emekli**

**olabilir, calisan erkekse 65 yasindan buyukse emekli olabilir**

**Eger (calisan kadinsa) {Kadin yasini kontrol et} ,**

**yoksa {erkek yasini kontrol et}**

**If (calisan kadinsa)**

**{if (yas>60) {emekli olabilirsin} else {emekli olamazsin}}**

**else**

**{if (yas>65) {emekli olabilirsin} else {emekli olamazsin}}**

**If Else Statements**

**Soru 11) Nested If kullanarak asagidaki soruyu cozen kodu yaziniz. Kullanicidan bir sifre girmesini isteyin**

**Eger ilk harf buyuk harf ise “A” olup olmadigini kontrol edin. Ilk harf A ise “Gecerli Sifre” degilse “Gecersiz Sifre” yazdirin.**

**Eger ilk harf kucuk harf ise “z” olup olmadigini kontrol edin. Ilk harf z ise “Gecerli Sifre” degilse “Gecersiz Sifre” yazdirin.**

**Soru12)Kullanıcıdan 4 basamakli bir sayi girmesini isteyin. Girdiği sayi 5’e bölünüyorsa son rakamını kontrol edin. Son rakamı 0 ise ekrana “5’e bölünen çift sayı” yazdırın. Son rakamı 0 değil ise “5’e bölünen tek sayı” yazdırın. Girdiği password 5’e bölünmüyorsa ekrana “Tekrar deneyin” yazdırın.**

**If Else If Statements Soru 13) Interview Question** 

**Kullanicidan artik yil olup olmadigini kontrol etmek icin yil girmesini isteyin.**

**Kural 1: 4 ile bolunemeyen yillar artik yil degildir**

**Kural 2: 4 ile bolunup 100 ile bolunemeyen yillar artik yildir**

**Kural 3: 4’un kati olmasina ragmen 100 ile bolunebilen yillardan sadece 400’un kati olan yillar artik yildir**

**https://app.diagrams.net/**