**1-) IDE ne anlama gelir ve neden IDE kullanmak iyi bir fikirdir?**

IDE "Entegre Geliştirme Ortamı" anlamına gelir ve yazılım geliştirme sürecinde kod yazmayı, hata ayıklamayı, derlemeyi ve test etmeyi sağlayan bir yazılım aracıdır.

**2-)Yeni bir nesne içeren "Happy Birthday!" kelimelerini içeren bir dize oluşturmak için hangi kodu yazarsınız? Bu yeni oluşturulan nesneye bir referansı text adıyla işaretlemek için bir kod yazın.**

Cevap: text = "Happy Birthday!"

**3-) Önceki soruda oluşturduğunuz string ifadeyi alıp, iki string içeren bir liste olarak ayırmak için hangi kodu yazarsınız? Bu yeni oluşturulan listeye bir referansı lst adıyla işaretlemek için bir kod yazın.**

Cevap: lst = text.split()

**4-) İkinci soruda oluşturduğunuz string ifadesini büyük harfe dönüştürmek için hangi kodu yazarsınız? Bu büyük harflere dönüştürülmüş yeni oluşturulan string ifadesine bDayWish adıyla bir referans atayın.**

Cevap:

bDayWish = text.upper()

**5-)Son üç soruya cevap vermek için yazdığınız kodları çalıştırsaydınız, bu üç kod satırının çalıştırılmasından sonra "text" adlı referansın işaret ettiği string ifadesi neyi içerecektir?**

Cevap: Önceki üç soruya verdiğimiz cevaplara dayanarak yazdığımız kodları çalıştırdıktan sonra "text" adlı referansın işaret ettiği string ifadesi, "Happy Birthday!" olacaktır.

İlk soruda "Happy Birthday!" ifadesi bir string olarak oluşturulmuş ve "text" adlı referansa atılmıştır. İkinci soruda "text" ifadesi büyük harflere dönüştürülmüş ve "bDayWish" adlı referansa atanmıştır. Son olarak, üçüncü soruda "text" ifadesi iki string içeren bir liste haline getirilmiş ve "lst" adlı referansa atanmıştır. Ancak bu değişiklikler "text" ifadesini değiştirmemiştir, hala aynı "Happy Birthday!" ifadesini içermektedir.

**6-)"Kent" kelimesini "Denise" ismiyle eşleştiren ve "Steve" kelimesini "Lindy" ismiyle eşleştiren bir sözlük nasıl oluşturulur? Bu iki durumda "Kent" ve "Steve" anahtarlar, "Denise" ve "Lindy" değerleridir.**

Cevap: sozluk = {"Kent": "Denise", "Steve": "Lindy"}

**7-) Önceki problemden farklı olarak, bir sözlük önceden oluşturulmadan bir anahtarın bir değere eşlenmesi için nasıl yapılır? İpuçları: Bu, ekteki belgedeki bir öğe ayarlama olarak adlandırılır. Boş bir sözlük oluşturmak ve ardından "Kent" kelimesini "Denise" ile eşleştirmek ve "Steve" kelimesini "Lindy" ile eşleştirmek için kısa bir kod yazın.**

Cevap:

sozluk = {}

sozluk["Kent"] = "Denise"

sozluk["Steve"] = "Lindy"

**8-) x < y yazıldığında hangi yöntem çağrılır? Bu yöntem hangi sınıfın bir üyesidir? Başka bir deyişle, bir programda x < y ifadesi ile karşılaştığınızda, x < y ifadesinin tam olarak ne anlama geldiğini anlamak için hangi sınıfa bakmanız gerektiğini nasıl bulursunuz?**

Cevap: x < y ifadesi yazıldığında, Python'da "**lt**" isimli özel bir karşılaştırma yöntemi çağrılır. Bu yöntem "object" sınıfının bir üyesidir ve varsayılan olarak tüm sınıflar tarafından kalıtım alınır. Eğer x ve y farklı sınıflardan nesneler ise, öncelikle x nesnesinin "**lt**" yöntemi çağrılır. Eğer "**lt**" yöntemi tanımlanmamışsa, nesnelerin varsayılan karşılaştırma davranışı kullanılır. Eğer x ve y aynı sınıfa ait nesneler ise, o sınıfın "**lt**" yöntemi çağrılır.

**9-)x << y yazıldığında hangi yöntem çağrılır?**

Cevap: Bu soruda, << işareti kullanılarak yapılan bir işlemin hangi yöntemi çağırdığı sorulmaktadır. Bu işlem "bitwise left shift" olarak adlandırılır ve "int" veri tipi için geçerlidir. Bu işlem sırasında "<<" operatörü kullanılır. Bu operatör, sol tarafındaki sayının ikili (binary) gösterimdeki bütün bitlerini, sağ tarafındaki sayı kadar sola kaydırır ve boş kalan bitleri 0 ile doldurur. Bu işlemin karşılığı olarak "int" veri tipindeki sınıfın "<<" isimli bir yöntemi bulunur.

**10-)"Loop and a half problem" nedir ve nasıl çözülür?**

Cevap: "Loop and a half problem", bir döngüde bir tam turun tamamlanmasına ek olarak bir yarım tur eklenmesi durumunda ortaya çıkan bir programlama hatasıdır. Bu sorunu çözmek için, döngü içindeki koşulları dikkatli bir şekilde kontrol etmek ve döngüyü sıfırlamak için gerekli olan tüm adımları belirlemek gerekir. Ayrıca, **break** veya **continue** ifadelerinin doğru bir şekilde kullanımı, döngüyü erken kesmenize veya atlamalar yapmanıza olanak tanır, böylece "loop and a half problem" engellenebilir.

**11-)Bir XML dosyasını okumak için döngü ve yarım çözümünü kullanmanız gerekiyor mu? Neden veya neden değil?**

Cevap:

Hayır, XML dosyasını okumak için döngü ve yarım çözümü kullanmak zorunda değilsiniz. Çünkü modern programlama dilleri ve kütüphaneleri, XML dosyalarını okumak için özel araçlar ve fonksiyonlar sağlarlar. Bu araçlar, XML verilerini okumak için döngü ve yarım gibi karmaşık teknikleri kullanmak yerine, verileri daha doğrudan ve kolay bir şekilde okumanıza olanak tanır. Örneğin, Python'da, ElementTree ve BeautifulSoup gibi kütüphaneler, XML dosyalarını kolayca okumanıza olanak sağlar.

**12-)Polimorfizm ve Operatör Aşırı Yükleme (Operator Overloading) kavramları birbirine yakından ilişkilidir. İki kavramın nasıl benzer olduğunu ve Python'un bunları nasıl desteklediğini kısaca açıklayabilir misiniz? (İPUCU: "Hayır" cevabı geçerli değildir.)**

Cevap: Polimorfizm, aynı işlemin farklı veri tipleri için farklı davranışlar sergilemesini sağlayan bir kavramdır. Yani aynı metodun, farklı sınıfların nesneleri için farklı şekillerde davranış sergilemesine izin verir. Python, polimorfizmi desteklemek için dinamik yazım özelliğini kullanır.

Operatör aşırı yükleme, operatörlere farklı davranışlar atamak için kullanılan bir kavramdır. Python, operatör aşırı yüklemeyi, herhangi bir sınıfın tanımlayabileceği belirli yöntemlerle destekler. Örneğin, '+' işlecinin farklı sınıflarda farklı şekillerde davranmasını sağlamak için, sınıfların '**add**' yöntemini tanımlaması gerekmektedir.

**13-)Bir programın kullanıcıdan bir tamsayı girmesini ve sonra kullanıcının girdiği tamsayıya kadar olan tüm çift tamsayıları toplamasını isterseniz ne yazarsınız? İPUCU: Bu işlemi gerçekleştirmek için range fonksiyonunun nasıl kullanılacağına göz atmanız ve ne tür bir programlama modeli kullanacağınıza karar vermeniz gerekebilir.**

Cevap:

n = int(input("Bir tamsayı girin: "))

toplam = 0

for i in range(2, n+1, 2):

toplam += i

print("2 ile", n, "arasındaki tüm çift sayıların toplamı:", toplam)

**14-)Tkinter kullanarak bir pencere nasıl oluşturulur?**

Tkinter ile bir pencere oluşturmak için öncelikle Tkinter modülünü projenize dahil etmeniz gerekir. Ardından bir ana pencere (root window) oluşturmak için Tk sınıfını kullanabilirsiniz.

**15-) Tkinter programında bir Frame nesnesinin amacı nedir?**

Cevap: Tkinter programında bir Frame nesnesi, diğer widget'ların düzenlenmesine yardımcı olan bir konteynerdir. Bir pencere üzerindeki widget'ları düzenlemek ve konumlandırmak için kullanılır. Frame, bir pencere içindeki widget'ların gruplandırılmasına olanak sağlar ve düzenli bir arayüz oluşturmak için kullanılır. Ayrıca, birden fazla widget'ı tek bir birim olarak ele almak ve özelliklerini birlikte ayarlamak için de kullanılabilir.

**16-)Tkinter çerçevesinde üç tür araç gereç (widget) nedir?**

Cevap:

Tkinter çerçevesindeki üç tür araç gereç (widget) şunlardır:

1. Temel araç gereçler (Basic widgets): Bu araç gereçler, pencere, düğme, etiket, metin kutusu, radyo düğmeleri, onay kutuları gibi kullanıcı arayüzü oluşturmak için temel olarak kullanılan araç gereçlerdir.
2. Dış görünüm araç gereçleri (Appearance widgets): Bu araç gereçler, kullanıcı arayüzünün dış görünümünü değiştirmek için kullanılır. Örneğin, pencereyi yeniden boyutlandırmak, kaydırma çubukları, ilerleme çubukları, diyalog kutuları gibi araçlar kullanılır.
3. Karmaşık araç gereçler (Complex widgets): Bu araç gereçler, temel araç gereçlerini birleştirerek daha karmaşık arayüzler oluşturmak için kullanılır. Örneğin, listeler, ağaçlar, tablolar, menüler gibi araçlar kullanılır.

**17-) Bir XML dosyasını okurken, dosyanın okunması için kaç satır kod gereklidir?**

Bir XML dosyasının okunması için kaç satır kod gerektiği dosyanın boyutuna, yapısına ve okuma işleminin nasıl yapılacağına bağlıdır. Basit bir XML dosyasını okumak için, dosyanın boyutuna, içeriğine ve amacına bağlı olarak birkaç satır veya onlarca satır kod yazmak gerekebilir.

**18-)Bir XML belgesinden tek bir öğeyi nasıl alabilirsiniz? Hangi satır (lar) kodunu yazmanız gerekiyor? Bir örnek verin.**

Cevap:

import xml.etree.ElementTree as ET

# XML belgesini açın ve ağaç yapısını oluşturun

tree = ET.parse('example.xml')

root = tree.getroot()

# İlk "book" etiketini bulun ve "title" etiketinin metnini yazdırın

book = root.find('book')

title = book.find('title').text

print(title)

**19-)XML belgesinde gezinirken, ondan bir öğe listesi nasıl alınır? Hangi satır(lar) kodu yazmanız gerekiyor? Bir örnek verin.**

Cevap: XML belgesinde gezinirken, belirli bir öğenin tüm alt öğelerine erişmek için **findall()** yöntemi kullanılır. Bu yöntem, bir XPath ifadesi ile birlikte kullanılabilir. XPath ifadesi, belgedeki hedef öğeleri belirlemek için kullanılır.

**20-)XML belgesinde bir nitelik (attribute) nedir ve bir niteliğin değerine nasıl erişilir? Metinden veya çevrimiçi bulduğunuz başka bir örnekte bir örnek verin.**

Cevap: XML belgesinde, bir öğenin nitelikleri onun özelliklerini belirten ek bilgilerdir. Örnek olarak, bir kitap öğesi şunları içerebilir: başlık, yazar, yayınevi, basım tarihi vb. Nitelikler, XML öğesinin başına yazılan bir etiket ile belirtilir ve etiketin içinde adı ve değeri yer alır. Niteliklere erişmek için, bir öğenin **attrib** özelliği kullanılır.

Örneğin, aşağıdaki XML belgesinde, **<book>** öğesi, **id**, **author**, **title**, **genre** ve **price** niteliklerine sahiptir:

<book id="bk101">

<author>Gambardella, Matthew</author>

<title>XML Developer's Guide</title>

<genre>Computer</genre>

<price>44.95</price>

<publish\_date>2000-10-01</publish\_date>

<description>An in-depth look at creating applications

with XML.</description>

</book>