



Oğuz Yarımtepe
LKD
oguzy (at) comu.edu.tr

Python ile Programlama



Oğuz Yarımtepe
oguzy (at) comu.edu.tr



Kapsam

- Hikayesi
- Eric'ten Tavsiyeler
- Yorumlar
- Dili Tanıyalım (Yapılabilecekler, söz dizimi, kuralları, v.s.)
- IDE, UDE
- Grafik Kütüphaneleri
- Nasıl Yardım Alırım?
- Dünyayı Kurtaracak Programın Yazımı



Hikayesi

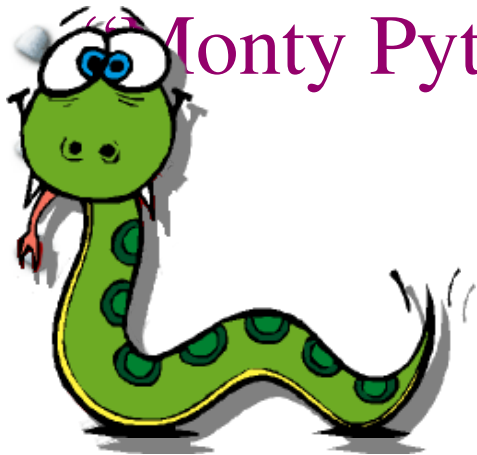


- Yaratıcısı Guido van Rossum.
- Lisansı 2.0 sürümünden itibaren GPL ile uyumlu hale getirildi



Hikayesi (devam)

Hollanda'da Centrum voor Wiskunde en Informatica (CWI) isimli araştırma enstitüsündeki Amoeba dağıtık işletim sistemi üzerinde çalışırken ABC dili yapısına benzeyen bir betik dili ve sistem yönetimi için de C'den veya kabuk betiklerinden daha etkin bir dile ihtiyaç duymasıyla 1989'da geliştirilmeye başlandı. 1991'de USENET'te duyuruldu.



Monty Python's Flying Circus ” (BBC komedi serisi)

Topluluk var (hızla gelişiyor*)

PSF (Python Software Foundation)

*Jython, IronPython, Nokia 60 serisi



Eric'ten Tavsiyeler

- Eric S. Raymond: Cathedral & Bazaar'ın yazarı, açık kaynak hareketinin kurucusu, Linus Torvalds ve Richard M. Stallman ile birlikte bu harekette adı geçen ilk 6 kişiden biri (kendi tabiri), Hacker'lık Sözlüğü'nün yazarı, pek çok açık kaynak yazılım, SSS ve web sayfasının sürdürücüsü



Nasıl Hacker Olunur? (ESR)

“Argo Dosyası'nda pek çok üstat (“hacker”) tanımlaması bulunmaktadır. Bunlar genellikle kod üstatlarını teknik beceri sahibi, problem çözmeden zevk alan ve sınırları aşan kişiler olarak tanımlarlar. Eğer nasıl üstat olunacağını öğrenmek istiyorsanız, bu tanımlardan sadece iki tanesi ilgi sahamızda olacaktır.

Uzman programcılar ve ağ sihirbazlarının, ilk zaman paylaşımlı mini bilgisayarlara ve en eski ARPAnet deneylerine kadar uzanan onlarca yıllık bir toplulukları ve ortak bir kültürleri vardır. “üstat” (“hacker”) kavramını bu kültürün üyeleri ortaya çıkarmışlardır. İnternet'i kuran, üstatlardır. Unix işletim sistemini bugünkü haline getirenler, üstatlardır. Halen Usenet'i çalışır halde tutanlar, üstatlardır. Webi ayakta tutanlar, üstatlardır. Eğer bu kültürün bir parçasıysanız, bu kültüre katkıda bulduysanız ve insanlar sizin kim olduğunuzu biliyor ve size “üstat” diye hitap ediyorsa, siz bir üstatsınız.

Üstatlık sadece yazılım konusu ile sınırlı değildir. Üstat düşünce yapısını elektronik veya müzik gibi diğer şeylere uygulayan insanlar vardır. Herhangi bir bilim veya sanat dalının en üst seviyelerinde de bunu görebilirsiniz. Yazılım üstatları farklı konularda çalışan bu yakın ruhları tanır ve onlara da üstat diyebilirler.”



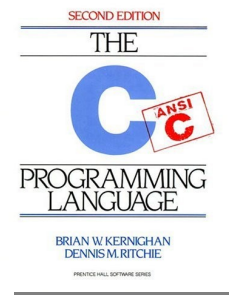
Nasıl Hacker Olunur? (devam)

- “Eğer bir programlama dili bilmiyorsanız, size Python ile başlamanızı öneririm. Python, tasarımı temiz, **belgelendirmesi*** iyi yapılmış ve yeni başlayanlara kolaylık tanıyan bir dildir. Öğrenebilecek ilk dil olarak iyi bir seçim olmasına rağmen, oyuncak değil, güçlü ve esnek bir dildir”
- “Java da programlamayı öğrenmek için iyi bir dildir. Python'dan daha zordur ama ürettiği kod daha hızlıdır. İkinci dil olarak öğrenebilecek en iyi dil olduğunu düşünüyorum.”
- “Eğer ciddi programlamaya girecekseniz, Unix'in ana dili olan C'yi öğrenmelisiniz.”
- * Kısaca değinelim, detayları sonraki slaytlarda



ve Kazanaan....

- “Makinaların güçlü olduğu günümüzde, makinanın çalışma verimliliği yerine sizin çalışma verimliliğinize öncelik tanıyan bir dil ile program yazmak daha akıllıca olacaktır. Sonuç olarak Python'u tercih etmenizi öneririm.”
- “En iyisi, bahsettiğim bu beş dilin (Python, Java, C/C++, Perl ve LISP) hepsini öğrenmektir.”





Yorumlar

Oğuz Yarımtepe
LKD
oguzy (at) comu.edu.tr

“Nesneye yönelik bir dil olması nedeni ile büyük ölçekli yazılımlar geliştirirken kolaylık sağlaması, çok hızlı prototip geliştirebilecek kadar geliştiriciyi ayrıntılardan yalıtıyor olması, hem betik için hem de çok büyük uygulamalar geliştirmek için uygun olması, geniş ve kararlı bir standart kütüphaneye sahip olması, temiz bir söz dizimi olması, kolay okunur ve kolay refactor edilebilir olması, dışardan geliştirmeye destek vermek isteyenlerin zorlanmadan adapte olabileceği bir modulariteyi yapısı gereği bünyesinde barındırması, bir çok kişinin biliyor olması, diğer diller ile kolayca genişletilebiliyor olması (örneğin c ile extend edip qt bindinglerinden yararlanıyoruz). İşte bunlar Pythonu Pardus'u geliştirirken tercih etmemizden bir kaçı.” A. Murat Eren, Uludağ Projesi, TUBITAK



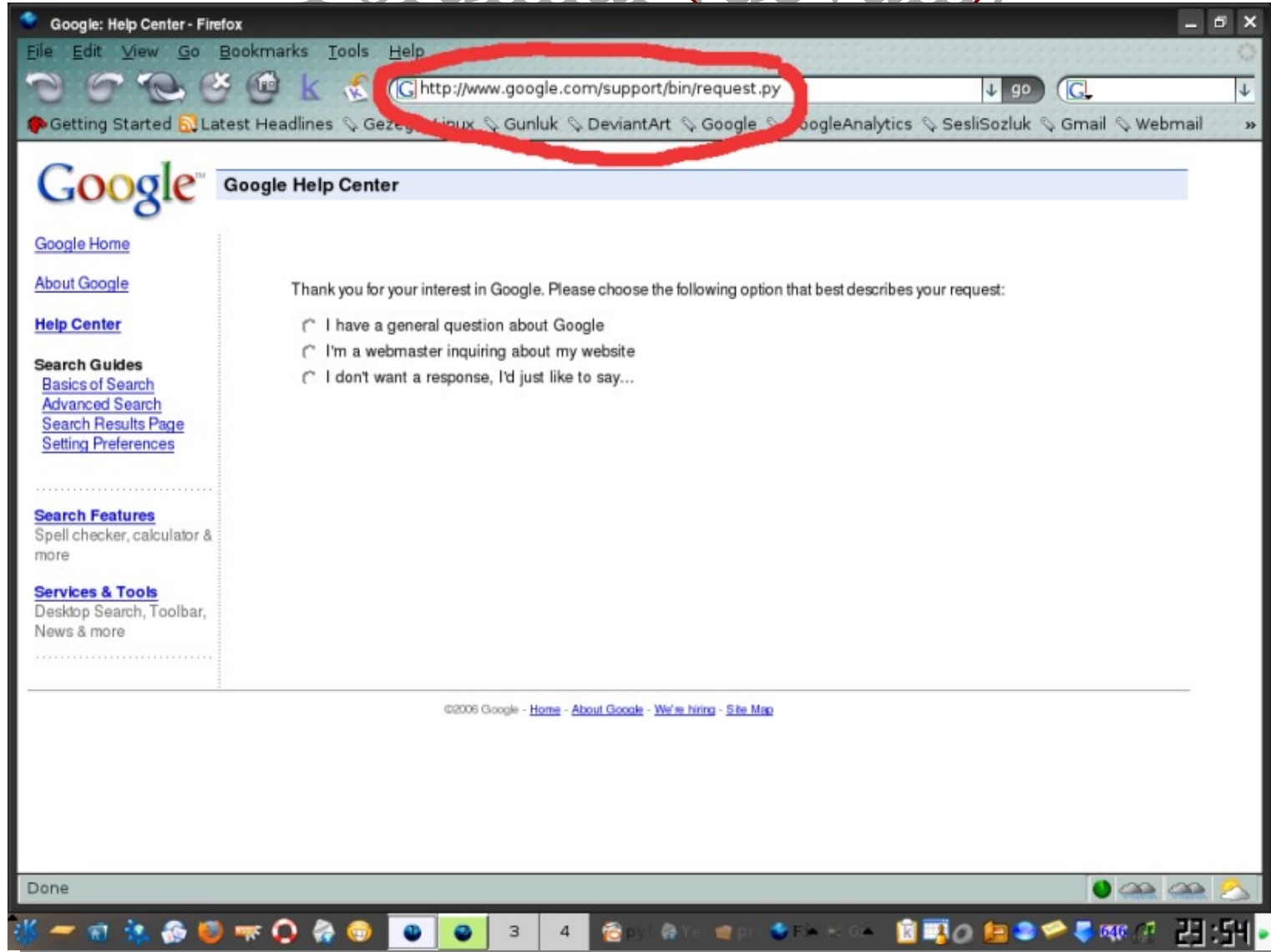
Yorumlar (devam)

Google

"Python has been an important part of Google since the beginning, and remains so as the system grows and evolves. **Today dozens of Google engineers use Python, and we're looking for more people with skills in this language.**" Peter Norvig, director of search quality at Google, Inc.



Yorumlar (devam)





Yorumlar (devam)

Industrial Light & Magic

"Python **plays a key role in our production pipeline**. Without it a project the size of **Star Wars: Episode II** would have been very difficult to pull off. From **crowd rendering to batch processing to compositing**, Python binds all things together," said Tommy Burnette, Senior Technical Director, Industrial Light & Magic.

"Python is everywhere at ILM. It's used to extend the capabilities of our applications, as well as providing the glue between them. Every CG image we create has involved Python somewhere in the process," Philip Peterson, Principal Engineer, Research & Development, Industrial Light & Magic.



Yorumlar (devam)

NASA

"NASA is using Python to implement a **CAD/CAE/PDM** repository and model management, integration, and transformation system which will be the core infrastructure for its next generation collaborative engineering environment. We chose Python because it provides **maximum productivity, code that's clear and easy to maintain, strong and extensive** (and growing!) libraries, and excellent capabilities for integration with other applications on **any platform**. All of these characteristics are essential for building **efficient, flexible, scalable**, and well-integrated systems, which is exactly what we need. Python has met or exceeded every requirement we've had," Steve Waterbury, Software Group Leader, NASA STEP Testbed.



Firaxis Games

Liste uzar gideeer...

"Like XML, scripting was extremely useful as both a mod tool and an internal development tool. **If you don't have any need to expose code and algorithms in a simple and safe way to others, you can argue that providing a scripting language is not worth the effort. However, if you do have that need, as we did, scripting is a no brainer, and it makes complete sense to use a powerful, documented, cross-platform standard such as Python.**"

"Python, like many good technologies, soon spreads virally throughout your development team and finds its way into all sorts of applications and tools. In other words, **Python begins to feel like a big hammer and coding tasks look like nails.**"

-- Mustafa Thamer of Firaxis Games, talking about **Civilization IV**. Quoted on page 18 of the August 2005 Game Developer Magazine.



Dili Tanıyalım

- Dinamik ve nesneye dayalı (kalıtım, çok biçimlilik, sarmalama, v.s.)
- Betik dili aynı zamanda
- Zengin bir kütüphane desteği ile gelir
- Birkaç günde öğrenilebilir
- Yazımı kolay, kısa, çabuk ve sonradan müdahalesi rahat
- Dünyayı kurtaracak program için ideal bir seçim :)



Web için Python

Web programlamada kullanılabilen geniş bir iskelet ve taslak sistemler var. Bir/birkaçı size çözüm getirir. En kötüsü Zope gibi bir sistem ile hemen herşeye sahip olabilirsiniz.

Bir kaçına bakalım.



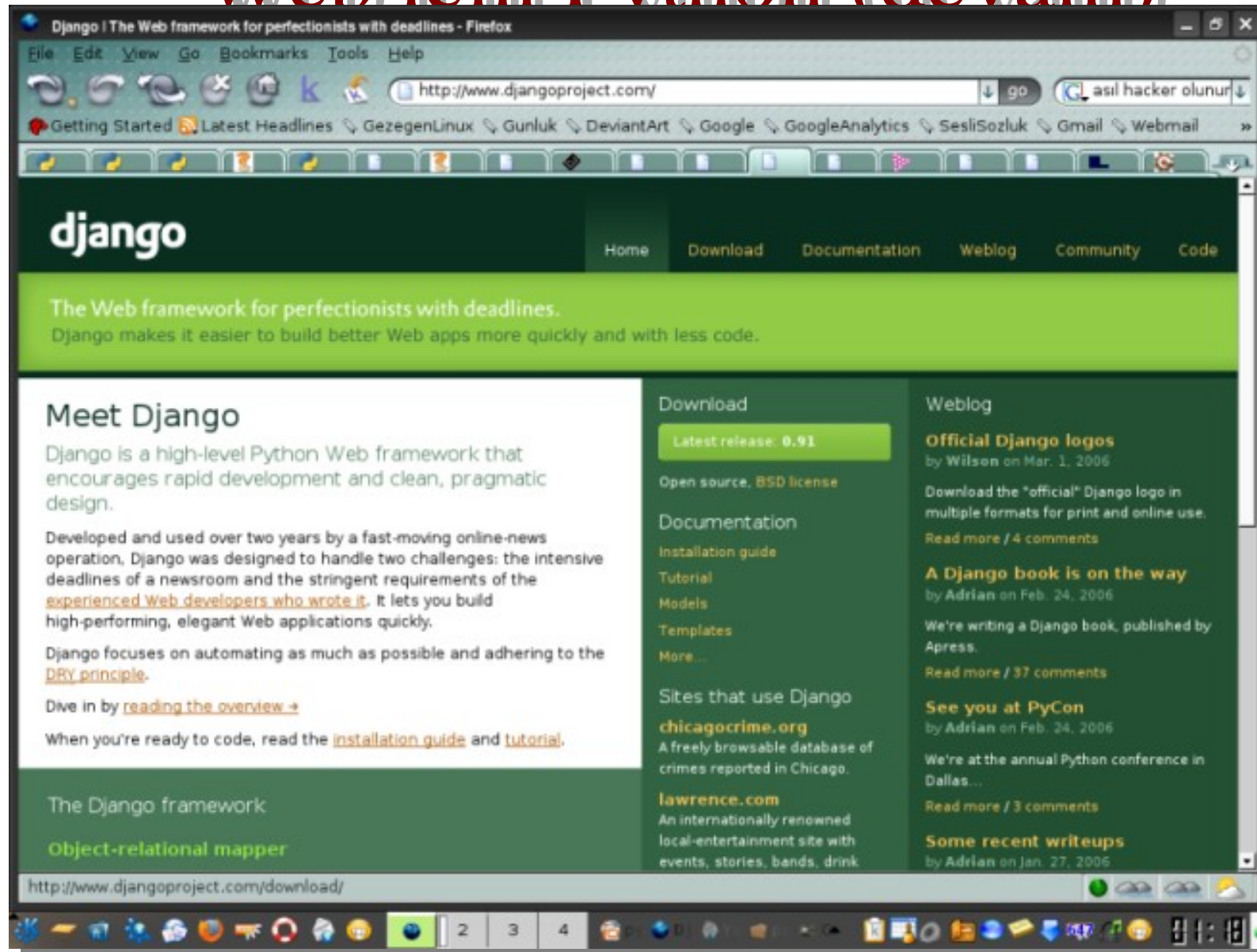
Web için Python (devam)

- Web sunucusu: TwistedMatrix
- Web iskeleti: **django**
 - Yüksek seviyeli Python web iskeleti oluşturma aracı:
 -
 - Nasıl bir şey?



Oğuz Yarıntepe
LKD
oguzy (at) comu.edu.tr

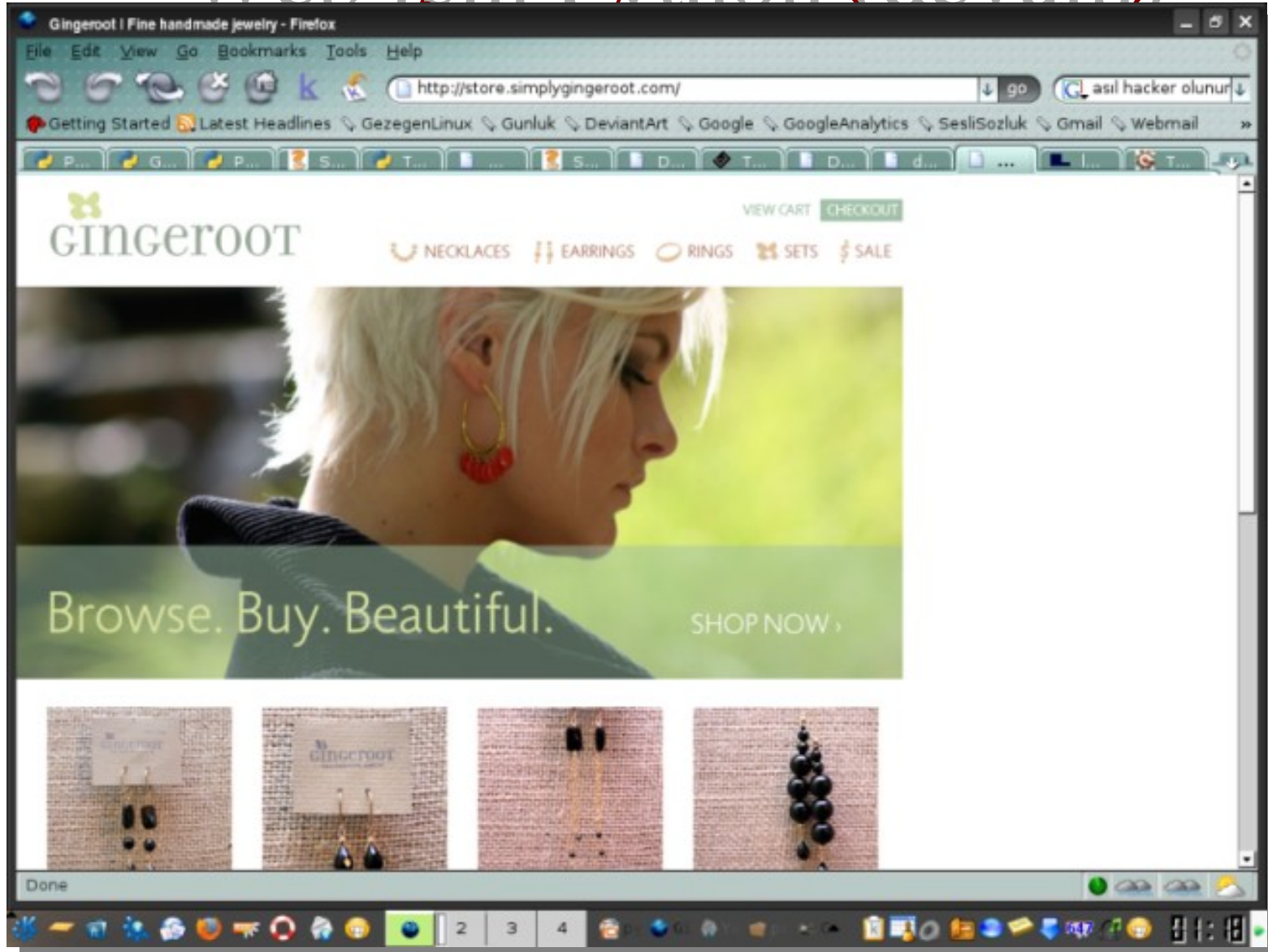
Web için Python (devam)





Oğuz Yarıntepe
LKD
oguzy (at) comu.edu.tr

Web için Python (devam)

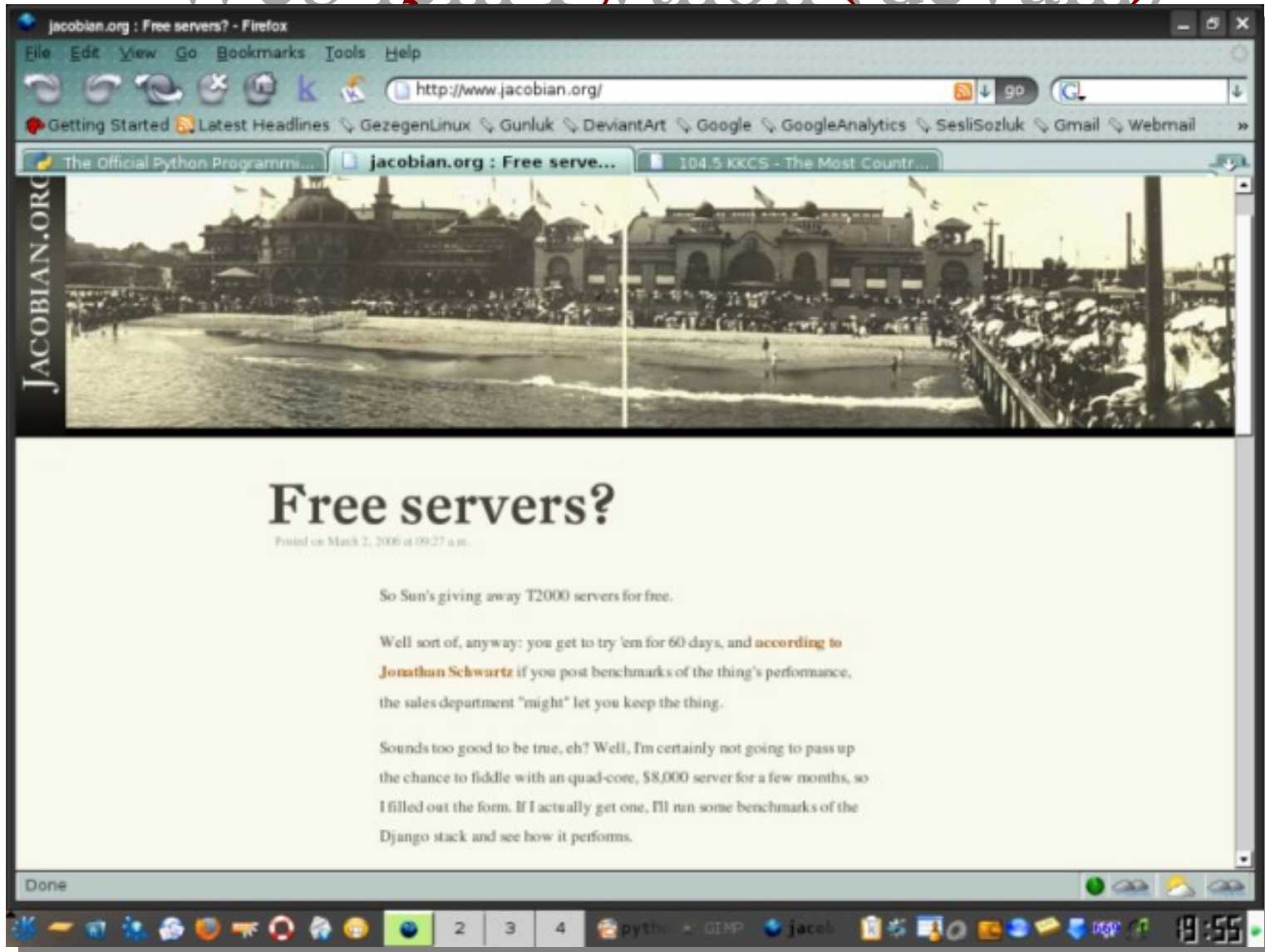




Oğuz Yarımtepe
LKD

oguzy (at) comu.edu.tr

Web için Python (devam)



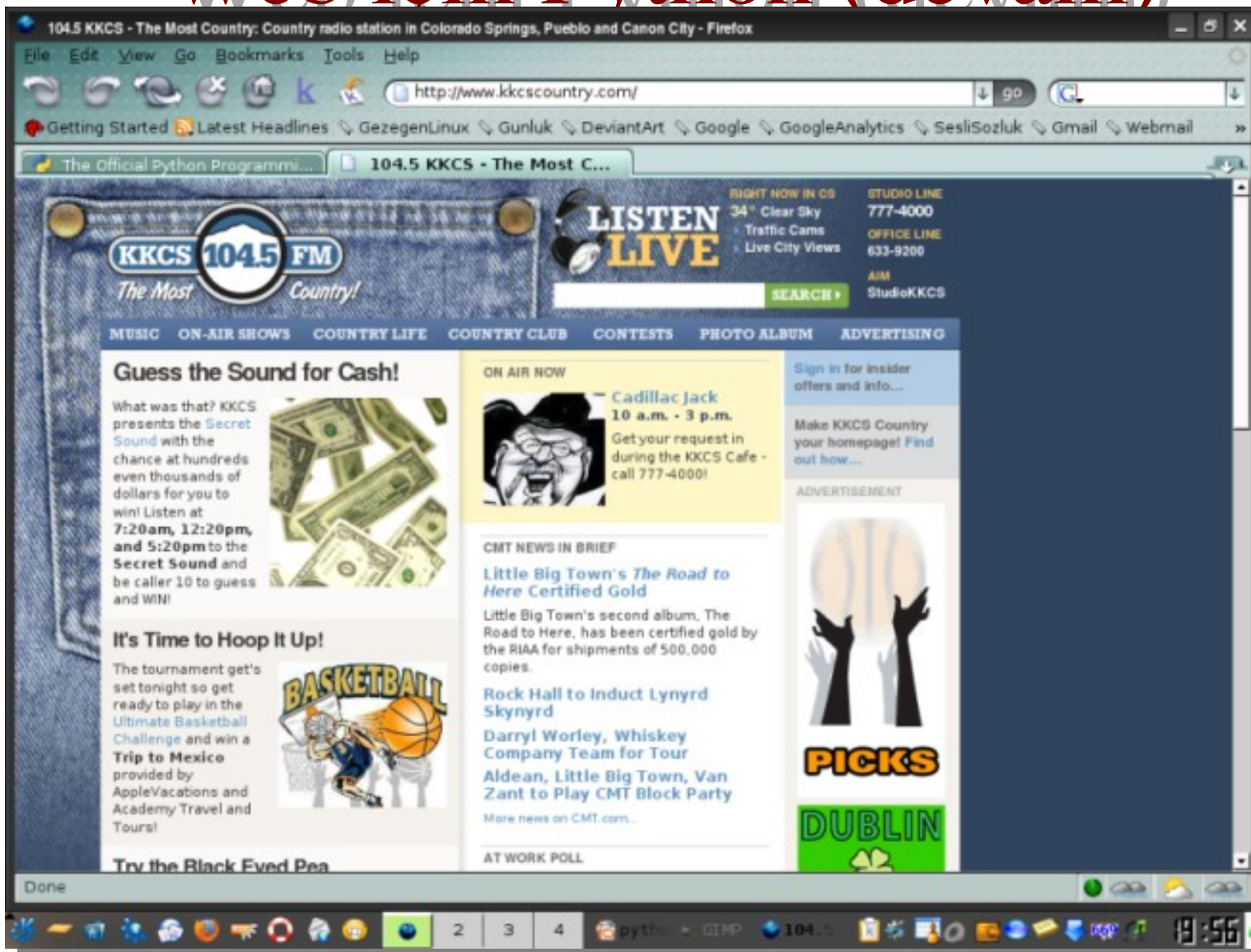


Oğuz Yarıntepe

LKD

oguzy (at) comu.edu.tr

Web için Python (devam)





Oğuz Yarımtepe
LKD

oguzy (at) comu.edu.tr

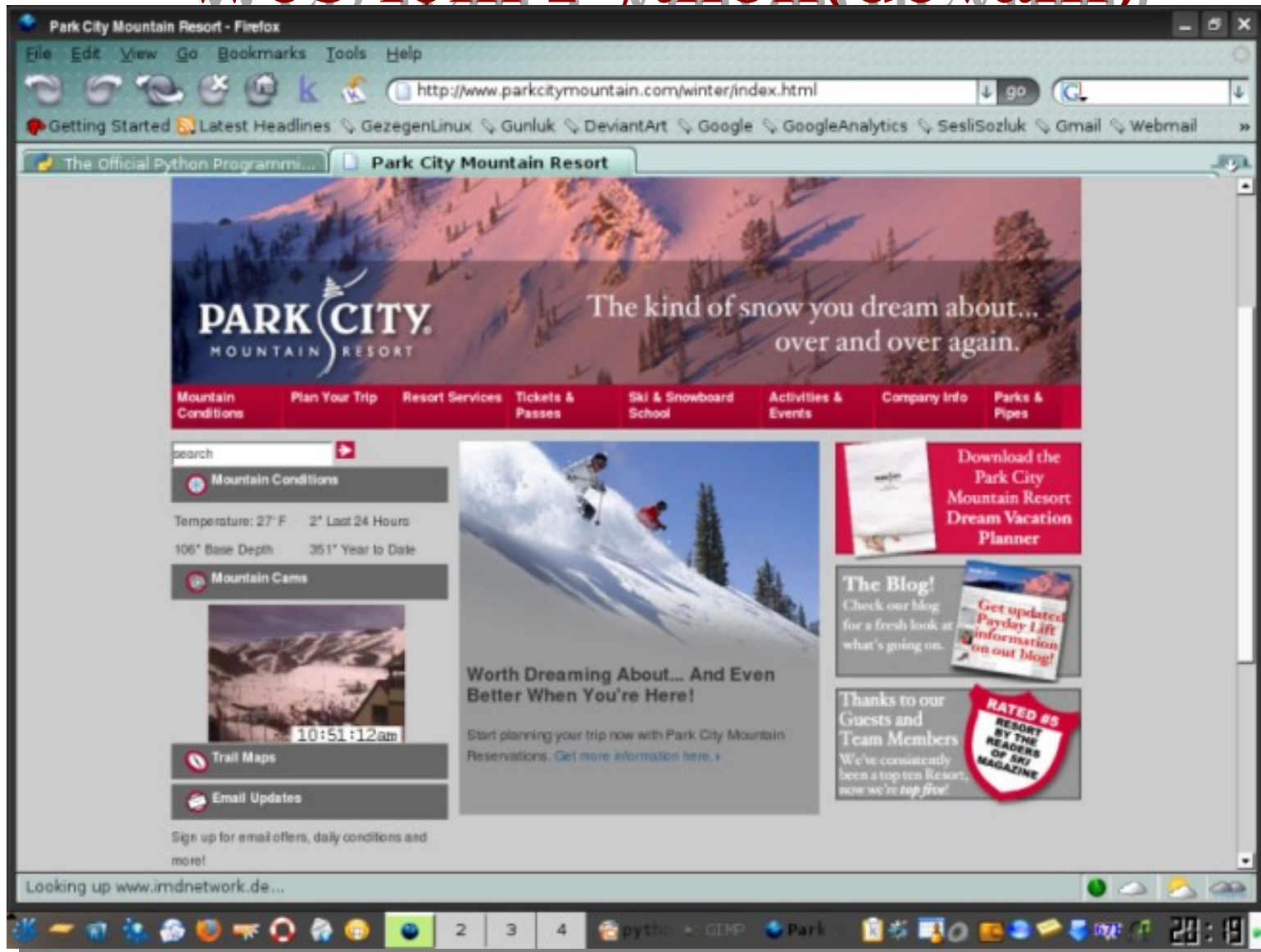
Web için Python (devam)

- Zope (Plone, CPS): Intranet, portal, özel uygulamalar için içerik yönetim sistemi.
- Kullananlar: Red Hat, NASA, NATO, US Navy, ... (zope)



Oğuz Yarıntepe
LKD
oguzy (at) comu.edu.tr

Web için Python (devam)





Web için Python (devam)

- XML, RSS (üretici, ayıklayıcı), e-posta, CGI ve diğer Internet protokolleri için zengin kütüphaneleri var.
- Diğerleri: ftp, pop3, imap, nntp, snmp, telnet, gopher (standart kütüphaneler içerisinde bulunurlar)
- Veri tabanlarıyla etkileşim mümkün: MySQL, Oracle, MS SQL Server, PostgreSQL, ...



Teknolojik Python

- Değişik ortamlarla uyum içerisinde çalışabilir:
 - **Jython**: Python'un %100 Java ile ifade edilmiş hali. Böylece Python'un herhangi bir Java ortamında çalışması mümkündür.
 - Java kullanıcıları Jython kütüphanelerini dahil ederek kullanıcıların betikler yazmalarını sağlayabilirler
 - Jython, Java ortamına karşılıklı etkileşim sağlar.
 - Programcıların pekçok Java ortamının hata ayıklamasına olanak sağlar
 - Python programları denkleri olan Java programlarından 2-10 kat daha hızlıdır.



Teknolojik Python (devam)

- **IronPython:** .Net ortamında çalışan Python. Etkileşimli bir yorumlayıcı sunar. Tüm .Net kütüphanelerinin Python içinde kullanımına olanak sağlar.
- **Windows?:** Windows COM ortamında çalışmayı sağlayan türevi de var.
- **Nokia 60 serisi:** N92, E70, 6630, N-Gage QD





Pilleri İçindedir!

- Standart kütüphanesiyle bile dünyayı kurtaran programı yazabiliriz :) :
- Çalışma zamanı servisleri:
- **sys**: Derleyicinin kendisi tarafından kullanılan nesne ve fonksiyonlara erişim
- **gc**: Çöp toplayıcına erişim
- ...



Pilleri İçindedir (devam)

- Karakter dizisi servisleri:
- **string**: Karakter dizileri üzerinde işlemler
- **re**: Düzenli ifadeler
- **unicodedata**: Unicode karakter veri tabanı
- ...



Pilleri İçindedir (devam)

- Genel işletim sistemi servisleri:
 - `os`: İşletim sistemi çalışırken gerekli rutinleri sağlar
 - `filecmp`: Dosyaları kıyasla
 - `popen2`: Komut çalıştırma
 - `datetime`: Zaman ile ilgili işler
 - `curses`: Konsolda grafik
 - ...



Pilleri İçindedir (devam)

- İşlerim Sistemi servisleri devam:
- **socket**: Socket işlemleri
- **zlib, gzip, bz2, zipfile**: Sıkıştırılmış dosya işlemleri
- **signal**: Sinyaller ile çalışma (süreçler arası)
- **thread**: Çoklu iş parçacıkları oluşturma
- ...



Pilleri İçindedir (devam)

• Internet Protokolleri:

• cgi, urllib, httplib, ftplib, poplib, imaplib, smtplib, telnetlib, xmlrpclib, ...

• Yapısal İşaretleme Dilleri:

• HTMLParser, sgmlib, htmllib, xmlib, ...

• Şifreleme Servisleri:

• md5, sha1

• Grafik Arayüzler:

• Tkinter, Tix, turtle, Idle,

• QT ve GTK için modüller kurulu gelmez



Pilleri İçindedir (devam)

•Diğer:

- **math**: C standartlarındaki matematiksel fonksiyonlara erişim
- **cmath**: Kompleks sayılarla işlemler
- **random**: Rasgele sayı üretme
- **calendar**: Tarih hesaplama ve yazdırma
- ...

Uçuşa Geçiyoruz

- Temin Etme ve Kurma
- Python Yorumlayıcısı ve Yorumlamaları
- Kontrol Blokları
- Listeler, Desteler, Sözlükler
- Haritalama ve Filtreler
- Fonksiyonlar
- Sınıflar
- İnişe Geçiş

Temin Etme ve Kurma

- <http://www.python.org/download/> adresinden Windows, Mac OS, Mac OS X, Linux, Sun Solaris, AS/400, Amiga, BeOS işletim sistemlerinden uygun olana göre program indirilir. Kurulur :).
- Linux: Dağıtıma göre rpm, deb, pisi paketi alınır kurulur. Bazı dağıtımlarla standart gelmektedir (FC3, Debian, Pardus)
- Kaynağı indirilir. configure, make, make install ile kurulum yapılır

Kurulum

```
localhost:~$ su -
```

```
Password: [enter your root password]
```

```
[root@localhost root]# wget
```

```
http://python.org/ftp/python/2.3/rpms/redhat-9/python2.3-2.3-5pydotorg
```

```
[root@localhost root]# rpm -Uvh python2.3-2.3-5pydotorg.i386.rpm
```

Kurulum (devam)

```
localhost:~$ su -
```

```
Password: [enter your root password]
```

```
localhost:~# apt-get install python
```

Kurulum (devam)

```
localhost:~$ su -
```

```
Password: [enter your root password]
```

```
localhost:~# wget
```

```
http://www.python.org/ftp/python/2.3/Python-2.3.tgz
```

```
localhost:~# tar xzf Python-2.3.tgz
```

```
localhost:~# cd Python-2.3
```

```
localhost:~/Python-2.3# ./configure
```

```
localhost:~/Python-2.3# make
```

```
localhost:~/Python-2.3# make install
```

Python Yorumlayıcısı

```
oguz@OzysDreams: /home/oguz - Kabuk - Konsole
Oturum Düzenle Görüntüle Yer İmleri Ayarlar Yardım
oguz@OzysDreams:~$ python
Python 2.3.5 (#2, Mar  6 2006, 10:12:24)
[GCC 4.0.3 20060304 (prerelease) (Debian 4.0.2-10)] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
>>> (98760001+9000000)*2^4
215520006
>>>
>>> 89/9.8
9.0816326530612237
>>>
>>> x = 9 + 4j
>>> y = -7
>>> x+y
(2+4j)
>>>
>>> x.imag
4.0
>>> x.real
9.0
>>>
>>> █
```

Python Yorumlayıcısı (devam)

```
oguz@OzysDreams: /home/oguz - Kabuk - Konsole
Oturum Düzenle Görüntüle Yer İmleri Ayarlar Yardım

>>> s = "iste size bir karakter dizisi"
>>> print (s)
iste size bir karakter dizisi
>>> print ("bu da benden")
bu da benden
>>>
>>> len(s)
29
>>> s[0]
'i'
>>> s[2]
't'
>>> s[-1]
'i'
>>> s[-5]
'i'
>>> s[1:5]
'ste '
>>> s[1:]
'ste size bir karakter dizisi'
>>> s[1:-5]
'ste size bir karakter d'
>>> s[-2:]
'si'
>>> █
```

Kontrol Blokları

```
oguz@OzysDreams: /home/oguz - Kabuk - Konsole
Oturum Düzenle Görüntüle Yer İmleri Ayarlar Yardım

>>> x = int(raw_input("Sayi giriniz: "))
Sayi giriniz: 97
>>> if x/2 == 0:
...     print ("cift sayi")
... elif x/2 != 0:
...     print ("tek sayi")
... else:
...     print ("baska bir sey!")
...
tek sayi
>>>
>>> dizi = ['elma', 'kiraz', 'muz']
>>> for x in dizi:
...     print x, len(x)
...
elma 4
kiraz 5
muz 3
>>>
>>> range(5)
[0, 1, 2, 3, 4]
>>> range(2,5)
[2, 3, 4]
>>> range(2,5,2)
[2, 4]
>>> █
```

Kontrol Blokları (devam)

```
oguz@OzysDreams: /home/oguz - Kabuk - Konsole
Oturum Düzenle Görüntüle Yer İmleri Ayarlar Yardım
>>> d = ["Ankara", "İstanbul", "İzmir", "İsparta"]
>>> for x in range(len(d)):
...     if (len(d[x]) > 5):
...         print (d[x])
...
Ankara
İstanbul
İsparta
>>>
>>> i = 0
>>> while(d[i] != "İsparta"):
...     print(d[i])
...     i += 1
...
Ankara
İstanbul
İzmir
>>> █
```

Listeler, Desteler, Sözlükler

- **Sözlük (dictionary):** Anahtar-Değer şeklinde bire bir eşleştirilerek tanımlanan eşleşmelerin olduğu dahili (built-in) veri yapısıdır
- $t = \{ \text{'anahtar1':deger1, 'anahtar2':deger2, ...} \}$
- İndeksleme anahtar (key) değerler ile yapılır
- Anahtarlar değiştirilemez veri yapılarından (karakter dizisi), sayılardan (int, float) veya destelerden oluşabilir.
- Sözlükleri *anahtar:değer* şeklinde ifadelerden oluşmuş sırasız bir dizi gibi düşünebiliriz.

Sözlükler

```
oguz@kiraz:~ - Kabuk No. 2 - Konsole
Oturum  Düzenle  Görüntüle  Yer imleri  Ayarlar  Yardım

>>> dahili = {}
>>> dahili = {"murat":1410, "kema1":1809, "cayci":1100}
>>> dahili["murat"]
1410
>>> dahili["murat"] = 1900
>>> dahili
{'cayci': 1100, 'murat': 1900, 'kema1': 1809}
>>>
>>> dahili["selma"] = 1243
>>> dahili
{'selma': 1243, 'cayci': 1100, 'murat': 1900, 'kema1': 1809}
>>>
>>> dahili.keys()
['selma', 'cayci', 'murat', 'kema1']
>>> dahili.values()
[1243, 1100, 1900, 1809]
>>> dahili.items()
[('selma', 1243), ('cayci', 1100), ('murat', 1900), ('kema1', 1809)]
>>>
>>> dahili.has_key('cayci')
True
>>> del dahili["cayci"]
>>> dahili
{'selma': 1243, 'murat': 1900, 'kema1': 1809}
>>>
>>> dahili.clear()
>>> dahili
{}
```

Sözlükler (devam)

- Üzerinde tanımlı bir sıralama metodu yok
- Yerleřtirmeler rasgele
- Her anahtardan sadece bir tane bulunur
- `d.has_values` diye bir Őey yok
- Gezinme (loop) teknikleri ile sözlüğü okumak mümkün

Sözlükler - Loop

```
oguz@kiraz:~ - Kabuk No. 2 - Konsole
Oturum  Düzenle  Görüntüle  Yer İmleri  Ayarlar  Yardım

>>> dahili = {"murat":1410, "kemal":1809, "cayci":1100}
>>>
>>>
>>>
>>> for x, y in dahili.iteritems():
...     print "adi:", x, " no:", y
...
adi: cayci  no: 1100
adi: murat  no: 1410
adi: kemal  no: 1809
>>>
>>> for x in dahili.itervalues():
...     print x
...
1100
1410
1809
>>> █
```

Listeler

- Dizi gibi düşünebiliriz, ama bilindik dizi kavramından fazlasını sunar
- Python içerisinde sıklıkla kullanılan bir veri yapısıdır
- Sıralı bir dizidir

Listeler (devam)

```
oguz@kiraz:~ - Kabuk No. 2 - Konsole
Oturum  Düzenle  Görüntüle  Yerimleri  Ayarlar  Yardım

>>> lst = ['a', 'b', 'c', 'd', 'e']
>>> lst
['a', 'b', 'c', 'd', 'e']
>>> lst[0]
'a'
>>> lst[4]
'e'
>>> lst[-1]
'e'
>>> lst[-5]
'a'
>>> lst[2:4]
['c', 'd']
>>> lst[2:]
['c', 'd', 'e']
>>> lst[:]
['a', 'b', 'c', 'd', 'e']
>>> lst[:-2]
['a', 'b', 'c']
>>>
>>> lst.append('f')
>>> lst
['a', 'b', 'c', 'd', 'e', 'f']
>>> lst.insert(1, 'ek')
>>> lst
['a', 'ek', 'b', 'c', 'd', 'e', 'f']
>>> lst.extend(["lseleman1", "lseleman2"])
>>> lst
['a', 'ek', 'b', 'c', 'd', 'e', 'f', 'lseleman1', 'lseleman2']
>>> lst.append('a')
>>> lst
['a', 'ek', 'b', 'c', 'd', 'e', 'f', 'lseleman1', 'lseleman2', 'a']
```

Listeler (devam)

```
oguz@kiraz:~ - Kabuk No. 2 - Konsole
Oturum  Düzenle  Görüntüle  Yer İmleri  Ayarlar  Yardım

>>> lst
['a', 'ek', 'b', 'c', 'd', 'e', 'f', 'lseleman1', 'lseleman2', 'a', 109, [2, 3, 4]]
>>> lst.append(109)
>>> lst
['a', 'ek', 'b', 'c', 'd', 'e', 'f', 'lseleman1', 'lseleman2', 'a', 109, [2, 3, 4], 109]
>>> lst2 = [2,3,4]
>>> lst.append(lst2)
>>> lst
['a', 'ek', 'b', 'c', 'd', 'e', 'f', 'lseleman1', 'lseleman2', 'a', 109, [2, 3, 4], 109, [2, 3, 4]]
>>>
>>> lst.index('a')
0
>>> lst.count('a')
2
>>> lst.remove(lst2)
>>> lst
['a', 'ek', 'b', 'c', 'd', 'e', 'f', 'lseleman1', 'lseleman2', 'a', 109, 109, [2, 3, 4]]
>>> lst.remove(109)
>>> lst
['a', 'ek', 'b', 'c', 'd', 'e', 'f', 'lseleman1', 'lseleman2', 'a', 109, 109]
>>> lst.sort()
>>> lst
[109, 109, 'a', 'a', 'b', 'c', 'd', 'e', 'ek', 'f', 'lseleman1', 'lseleman2']
>>> lst.reverse()
>>> lst
['lseleman2', 'lseleman1', 'f', 'ek', 'e', 'd', 'c', 'b', 'a', 'a', 109, 109]
>>> lst.pop()
109
>>> lst
['lseleman2', 'lseleman1', 'f', 'ek', 'e', 'd', 'c', 'b', 'a', 'a', 109]
>>> █
```

Listeler (devam)

```
oguz@kiraz:~ - Kabuk No. 2 - Konsole
Oturum  Düzenle  Görüntüle  Yer İmleri  Ayarlar  Yardım
>>>
>>>
>>>
>>> liste = [1,2,3]
>>>
>>> liste *= 2
>>>
>>> liste
[1, 2, 3, 1, 2, 3]
>>>
>>> k = [4,5,6]
>>>
>>> liste += k
>>>
>>> liste
[1, 2, 3, 1, 2, 3, 4, 5, 6]
>>>
>>> █
```

Desteler (Tuple)

- Desteler deđiřtirilemez listelerdir
- Listelerle aynı özelliklere sahiptir, fakat elemanları sabittir (ekleme, silme, sıralama yapılamaz)

Desteler (devam)

```
oguz@kiraz:~ - Kabuk No. 2 - Konsole
Oturum  Düzenle  Görüntüle  Yer İmleri  Ayarlar  Yardım
>>> t = (1,3,5,7)
>>> t
(1, 3, 5, 7)
>>> t[0]
1
>>> t[-1]
7
>>> t[2:-1]
(5,)
>>> t.append(4)
Traceback (most recent call last):
  File "<stdin>", line 1, in ?
AttributeError: 'tuple' object has no attribute 'append'
>>> █
```

Desteler (devam)

- Listelerden daha hızlı
- “yazma-korumalı” bir tanımlama için kullanılabilir
- İşin başında tanımlıyıp sonrasında da değişmeyecek bir liste için deste kullanmak performans artışı sağlar (gezinme (iteration) daha hızlı)

Haritalama (map) ve Filtreler

```
oguz@kiraz:~ - Kabuk No. 2 - Konsole
Oturum  Düzenle  Görüntüle  Yer İmleri  Ayarlar  Yardım
>>>
>>>
>>> li = [1,2,3,4]
>>>
>>> [x*3 for x in li]
[3, 6, 9, 12]
>>>
>>> [x*3 for x in li if x%2 != 0]
[3, 9]
>>>
>>> li
[1, 2, 3, 4]
>>> █
```

Fonksiyonlar

```
oguz@kiraz:~ - Kabuk No. 2 - Konsole
Oturum  Düzenle  Görüntüle  Yer imleri  Ayarlar  Yardım

>>> def fib2(n): # Fibonacci serisi
...     """Fibonacci dizisi donduren fonksiyon"""
...     result = []
...     a, b = 0, 1
...     while b < n:
...         result.append(b)    # eksin burada
...         a, b = b, a+b
...     return result
...
>>> t = fib2(100)
>>> t
[1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]
>>>
>>> def f(x, y=0, z="isim"):
...     return (str(x)+str(y)+z)
...
>>> f(1)
'10isim'
>>> █
```

Sınıflar

```
oguz@kiraz:~ - Kabuk No. 2 - Konsole
Oturum  Düzenle  Görüntüle  Yer İmleri  Ayarlar  Yardım

>>> class Dizi:
...     def __init__(self):
...         self.data = []
...     def add(self, x):
...         self.data.append(x)
...     def addtwice(self, x):
...         self.add(x)
...         self.add(x)
...
>>> x = Dizi()
>>> x.add(3)
>>> x.addtwice(4)
>>> x.data
[3, 4, 4]
>>> █
```

İniş Geçiyoruz

- IDE – Integrated Development Environment
- Linux: Eric, DrPython, Anjuta, Eclipse, Kdevelop, IDLE
- Windows: ActiveGrid, Komodo, PyScripter, PythonWin, PythonWorks, Visual Python,
- <http://wiki.python.org/moin/IntegratedDevelopmentEnvironments>
-
- UDE – Unintegrated Development Environment
- vim, emacs

Kemerlerinizi Çözebilirsiniz

- **IPython**
- Gelişmiş Etkileşimli Python Kabuğu
- Python standart kabuğu + Kılavuz sayfaları
- Kullandığım modulün metodları neler, nasıl kullanılır, join fonksiyonu diye bir şey vardı nasıl kullanılıyordu bu, yeni bir modül yükledim kaynak koduna bakabilir miyim, ...

Ipython'dan İnci Taneleri

```
oguz@kiraz:~ - Kabuk No. 2 - Konsolle
Oturum  Düzenle  Görüntüle  Yer İmleri  Ayarlar  Yardım

oguz@kiraz ~ $ ipython
Python 2.4.2 (#1, Jan 25 2006, 00:43:19)
Type "copyright", "credits" or "license" for more information.

IPython 0.6.15 -- An enhanced Interactive Python.
?      -> Introduction to IPython's features.
%magic -> Information about IPython's 'magic' % functions.
help   -> Python's own help system.
object? -> Details about 'object'. ?object also works, ?? prints more.

In [1]: import string

In [2]: string.
string.Template          string._re                string.ljust
string._TemplateMetaClass string.ascii_letters      string.lower
string.__class__         string.ascii_lowercase   string.lowercase
string.__delattr__       string.ascii_uppercase   string.lstrip
string.__dict__          string atof               string.maketrans
string.__doc__           string.atof_error        string.octdigits
string.__file__          string.atoi               string.printable
string.__getattr__       string.atoi_error        string.punctuation
string.__hash__          string.atol               string.replace
string.__init__          string.atol_error        string.rfind
string.__name__          string.capitalize         string.rindex
string.__new__           string.capwords           string.rjust
string.__reduce__        string.center             string.rsplit
string.__reduce_ex__     string.count              string.rstrip
string.__repr__          string.digits             string.split
string.__setattr__       string.expandtabs         string.splitfields
string.__str__           string.find               string.strip
string._float            string.hexdigits          string.swapcase
string._idmap            string.index              string.translate
```

lpython (devam)

```
oguz@kiraz:~ - Kabuk No. 2 - Konsole
Oturum  Düzenle  Görüntüle  Yer İmleri  Ayarlar  Yardım

In [3]: string.lower
string.lower      string.lowercase

In [3]: string.lower?
Type:             function
Base Class:      <type 'function'>
String Form:     <function lower at 0xa7b7bfb4>
Namespace:      Interactive
File:            /usr/lib/python2.4/string.py
Definition:     string.lower(s)
Docstring:
    lower(s) -> string

    Return a copy of the string s converted to lowercase.

In [4]: █
```

lpython (devam)

```
oguz@kiraz:~ - Kabuk No. 2 - Konsole
Oturum  Düzenle  Görüntüle  Yer imleri  Ayarlar  Yardım

In [5]: st
staticmethod  str          string

In [5]: string.lower??
Type:         function
Base Class:   <type 'function'>
String Form:  <function lower at 0xa7b7bfb4>
Namespace:   Interactive
File:        /usr/lib/python2.4/string.py
Definition:  string.lower(s)
Source:
def lower(s):
    """lower(s) -> string

    Return a copy of the string s converted to lowercase.

    """
    return s.lower()

In [6]:
```



Grafik Kütüphaneleri

- TK: <http://wiki.python.org/moin/TkInter>
- wxWidgets: <http://www.wxpython.org/>
- QT: <http://www.riverbankcomputing.co.uk/pyqt/>
- GTK: <http://www.pygtk.org/>
- Delphi: <http://mmm-experts.com/Products.aspx?ProductId=3>



Nereden Yardım Alabilirim?

- <http://www.python.org> (e-posta listeleri, kılavuzlar, vs.)
- <http://www.python.org/doc/>
- <http://www.turkpython.net/> : Türkçe Python Portalı
- <http://www.belgeler.org/uygulamalar/python-tutorial.html>
: Python Kılavuzu
- Ipython :)



Oğuz Yarımtepe
LKD
oguzy (at) comu.edu.tr

Dünyayı Kurtaran Program Yazımı

:)