

# Poglavlje 1

## Prvi koraci: program Game Over

Sušтина programiranja svodi se na to da učinite da računar radi ono što vi zadate. To nije najbolja tehnička definicija programiranja, ali je prilično tačna. Pošto savladate Python, znaćete da napišete program, bez obzira na to da li je u pitanju jednostavna igra, kratak pomoćni program ili poslovna aplikacija s kompletnim grafičkim korisničkim interfejsom. To će biti potpuno vaše delo, nešto što ste vi napravili, a radiće tačno ono što ste vi odredili. Programiranje je delimično nauka, delimično umetnost, a uvek je velika avantura. Ovim poglavljem započinjete svoje putovanje u oblast programiranja na jeziku Python. Tu ćete saznati:

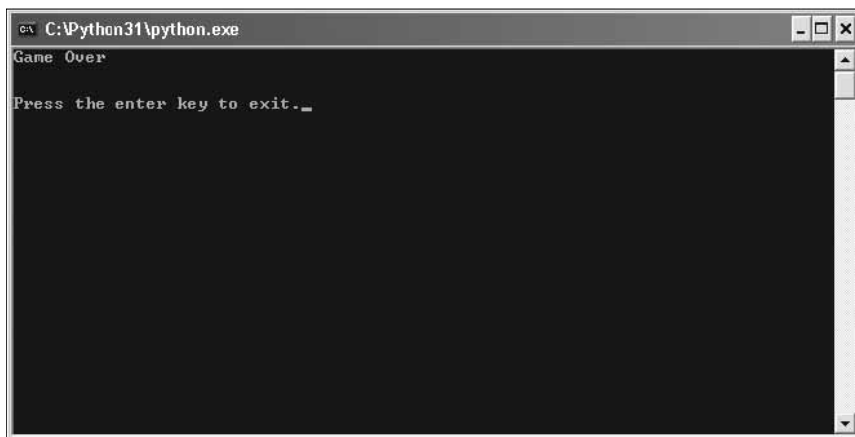
- šta je Python i po čemu je tako izuzetan
- kako ćete instalirati Python na svoj računar
- kako ćete prikazati tekst na ekranu
- šta su komentari i kako se koriste
- kako se u Pythonovom razvojnom okruženju programi pišu, menjaju, izvršavaju i snimaju na disk

### Pregled igre Game Over

Projekat iz ovog poglavlja, Game Over, prikazuje dve najomrznutije reči u svetu računarskih igara: „Game Over“ (igra je završena). Slika 1-1 prikazuje program u toku rada.

Slika 1-1 prikazuje ono što se zove *prozor konzole* (engl. *console window*), u kome se prikazuje isključivo tekst. Mada ti prozori nisu tako dopadljivi kao prozori u grafičkom korisničkom interfejsu (engl. *Graphical User Interface, GUI*), konzolske aplikacije se lakše pišu i dobra su osnovica od koje bi programer početnik mogao da krene.

Program Game Over veoma je jednostavan; to je zapravo jedan od najjednostavnijih programa koje možete napisati na jeziku Python – zato je i predstavljen



Slika 1-1: Dobro poznate reči iz računarskih igara.

u ovom poglavlju. Pisanje jednog tako skromnog programa omogućava vam da obavite celu pripremu koja je neophodna kako biste počeli da programirate na jeziku Python, kao što je instaliranje samog jezika na vaš sistem. Proći ćete i kroz ceo postupak pisanja, snimanja na disk i pokretanja programa. Pošto savladate te osnove, bićete spremni da se upustite i u složenije programe koji zahtevaju ozbiljniji trud.

---

## U STVARNOM SVETU

Program Game Over samo je varijanta tradicionalnog programa Hello World, koji na ekranu prikazuje reči „Hello World“. Program Hello World često je prvi program koji početnik napiše kao svoj prvi korak u novom jeziku. To je toliko uobičajen prvi program, da je Hello World postao uobičajen izraz u programiranju.

---

## Uvod u Python

Python je moćan i lako razumljiv jezik koji je smislio Guido van Rossum, a prva verzija je objavljena 1991. godine. Python omogućava da brzo napišete jednostavan projekat. Ali Python se lepo skalira i može se upotrebiti i za pisanje važnih komercijalnih aplikacija.

Ako proučite bilo koju raspoloživu dokumentaciju o Pythonu, možda ćete uočiti zabrinjavajući broj pominjanja reči „spam“, „eggs“ (jaja) i broja 42. Sva ta pominjanja izrazi su poštovanja prema engleskoj komičarskoj trupi Monty Python, po kojoj je jezik dobio ime. Upros tome što mu je Guido van Rossum nadevao ime Python po toj trupi, zvanična maskota jezika postala je zmija piton. (Što je pametna ideja, jer bi bilo prilično teško smestiti lica svih šest britanskih komičara na ikonicu programa.)

Postoji mnogo programskih jezika. Po čemu je Python tako izuzetan? Dozvolite mi da vam objasnim.

## Python se lako koristi

Najvažniji cilj svakog programskog jezika jeste da premosti jaz između programerovog mozga i računara. Većina popularnih jezika za koje ste možda čuli, kao što su Visual Basic, C# i Java smatra se *jezicima visokog nivoa*, što znači da su bliži govornim jezicima nego mašinskom jeziku. A oni to zaista i jesu. Ali Python, sa svojim jasnim i jednostavnim pravilima, još je bliži engleskom govornom jeziku. Pisanje programa na jeziku Python toliko je jednostavno da je nazvano „programiranje brzinom razmišljanja“. Posledica jednostavne upotrebe Pythona jeste veća produktivnost profesionalnih programera. Programi na jeziku Python kraći su i pišu se za manje vremena nego programi na mnogim drugim popularnim jezicima.

## Python je moćan

Python ima svu moć koju biste i očekivali od savremenog programskog jezika. Kada dođete do kraja ove knjige, bićete u stanju da pište programe koji rade s grafičkim korisničkim interfejsom, obrađuju datoteke i rade s raznim vrstama struktura podataka.

Python je dovoljno moćan da privlači programere širom sveta, i kompanije kao što su Google, IBM, Industrial Light + Magic, Microsoft, NASA, Red Hat, Verizon, Xerox i Yahoo!. Python koriste kao alat i profesionalni programeri igara. Kompanije Electronic Arts, 2K Games i Disney Interactive Media Group objavljuju igre u koje je ugrađen Python.

## Python je objektno orijentisan

*Objektno orijentisano programiranje* (OOP) savremen je pristup rešavanju problema pomoću računara, čija je suština intuitivna metoda predstavljanja informacija i akcija u programu. To svakako nije jedini način pisanja programa, ali je za obimne projekte često najbolji.

Jezici kao što su C#, Java i Python, objektno su orijentisani. Ali Python pruža i nešto još bolje. U jezicima C# i Java, primena OOP-a nije pitanje izbora. Zbog toga su kratki programi nepotrebno složeni, pa su programeru početniku neophodna prilično opsežna objašnjenja pre nego što bude u stanju da napravi bilo šta vredno pomena. Python primenjuje drugačiji pristup. U Pythonu, upotreba OOP tehnika nije obavezna. Cela OOP snaga stoji vam na raspolaganju, ali je možete iskoristiti samo kada vam zatreba. Imate jednostavan programčić za koji vam ne treba OOP? Nema problema. Imate opsežan projekat na kojem radi ekipa programera i za koji je potreban OOP? Može i tako. Python pruža i moć i fleksibilnost.

## Python je jezik koji se ponaša kao „lepak“

Python se može integrisati s jezicima kao što su C, C++ i Java. To znači sledeće: kada radi s Pythonom, programer može iskoristiti rezultate nečeg već napravljenog na nekom drugom jeziku. To takođe znači da on ili ona može iskoristiti prednosti koje pružaju drugi jezici, kao što je ubrzanje koje C ili C++ mogu da pruže, a da pri tome uživa u lakoći pisanja programa koja je zaštitni znak programiranja na jeziku Python.

## Python radi svuda

Python radi na svačemu – od ručnih računara do Craya. A ako slučajno nemate superračunar u svom podrumu, Python možete koristiti i na mašinama koje rade pod Windowsom, Macintoshem ili Linuxom. A to su samo prve stavke na spisku.

Programi napisani na Pythonu *ne zavise od platforme*, što znači sledeće: bez obzira na operativni sistem pod kojim pišete program, on će raditi pod svakim drugim operativnim sistemom u kojem postoji Python. Ako program napišete na svom PC-ju, kopiju možete poslati e-poštom prijatelju koji ima Linux ili svojoj tetki koja ima Mac i program će raditi (pod uslovom da vaš prijatelj i tetka imaju instaliran Python na svojim računarima).

## Python podržava jaka zajednica korisnika

Za većinu programskih jezika postoje organizovane grupe korisnika na Internetu, ali Python ima i nešto što se zove lista slanja Python Tutor, što je manje formalan način na koji programeri početnici mogu postavljati svoja prva pitanja. Ta lista se nalazi na adresi <http://mail.python.org/mailman/listinfo/tutor>. Uprkos tome što je ime liste Tutor, svako, bez obzira na to da li je početnik ili stručnjak, može odgovarati na pitanja.

Postoje i druge zajednice korisnika Pythona usredsređene na druge oblasti, ali je svima zajedničko to da su prijateljski nastrojene i otvorene. To je sasvim razumljivo pošto je sam jezik toliko jednostavan za početnike.

## Python je besplatan i otvorenog koda

Python je besplatan. Možete ga instalirati na svoj računar a da vas to nikad ne košta ni pare. Ali, Pythonova licenca vam omogućava i mnogo više od toga. Python možete kopirati i menjati. Možete ga čak i dalje prodavati ako želite (ali nemojte još davati otkaz na svom tekućem poslu). Prihvatanje ideala otvorenog koda sastavni su deo onoga što Python čini tako popularnim i uspešnim.

## Postavljanje Pythona na Windows

Pre nego što se upustite u pisanje svog prvog programa na jeziku Python, potrebno je da instalirate taj jezik na svoj računar. Ali nemojte brinuti – vodiću vas kroz postupak instaliranja Pythona na računar koji radi pod Windowsom.

## Instaliranje Pythona na Windows

Da biste instalirali Python na Windows, uradite sledeće:

1. Preuzmite instalacionu datoteku Pythona za Windows, koju ćete naći na Web lokaciji pridruženoj ovoj knjizi ([www.courseptr.com/downloads](http://www.courseptr.com/downloads)). Ta datoteka se nalazi u direktorijumu Software, u poddirektorijumu Python; ime datoteke je `python-3.1.msi`.
2. Pokrenite instalacionu datoteku Pythona za Windows, `python-3.1.msi`. Slika 1-2 prikazuje instalacioni program u radu.



Slika 1-2: Vaš računar će uskoro postati domaćin Pythonu.

3. Prihvatite ponuđenu konfiguraciju. Kada završite, imaćete Python 3.1 na svom sistemu.



Web lokacija koja je pridružena ovoj knjizi nalazi se na adresi [www.coursetr.com/downloads](http://www.coursetr.com/downloads). Tu ćete naći kôd svih programa koji se pominju na stranicama knjige, zajedno sa svim potrebnim pratećim datotekama i instalacionim programima. Detaljniji opis šta je sve na raspolaganju naći ćete u prilogu A, „Prateća Web lokacija“.

## Postavljanje Pythona na druge operativne sisteme

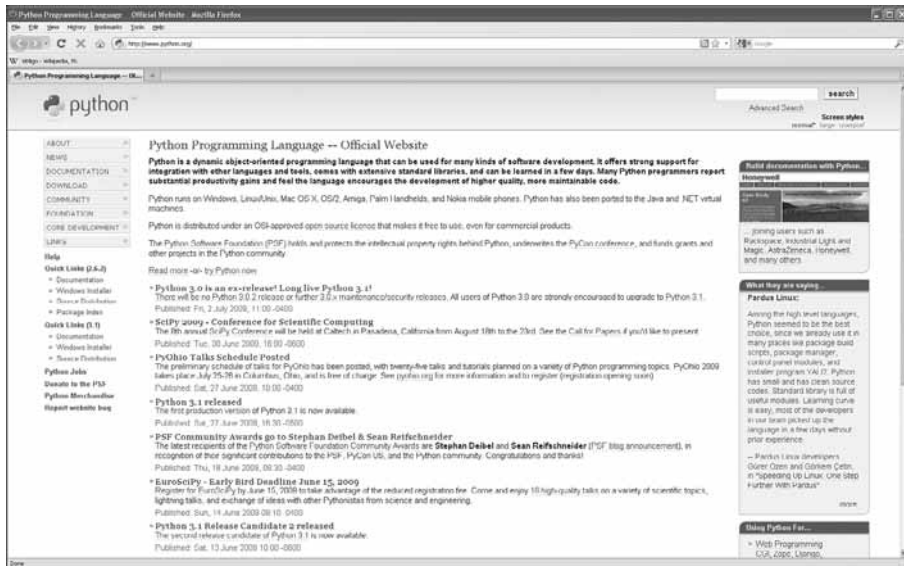
Python radi na doslovno više desetina operativnih sistema. Ako koristite nešto što nije Windows, obavezno posetite zvaničnu Web lokaciju jezika Python, <http://www.python.org>, da biste preuzeli najnoviju verziju jezika za vaš računar. Izgled početne stranice Pythona prikazan je na slici 1-3.



Možda je na vašem računaru već instaliran Python; ipak, potrebna vam je verzija Pythona 3.1 da bi programi iz ove knjige radili ispravno.

## Uvod u IDLE

Uz Python se isporučuje i integrisano razvojno okruženje pod imenom IDLE. **Razvojno okruženje** se sastoji od skupa alatki koje olakšavaju pisanje programa. Možete ga zamisliti kao program za obradu teksta, s tim što ne služi za tekst nego za programe. Ali, IDLE nije samo mesto gde ćete pisati, snimati na disk i menjati svoje programe, već ima i dva radna režima: interaktivni i skript.



Slika 1-3: Posetite Web lokaciju jezika Python da biste preuzeli najnoviju verziju Pythona i pročitali gomile relevantnih informacija.

## Programiranje u interaktivnom režimu

Najzad, došlo je vreme da malo „isprljate ruke“ tako što ćete nešto zaista isprogramirati na jeziku Python. Rad u Pythonu najbrže ćete započeti u interaktivnom režimu. U tom režimu možete narediti Pythonu šta da uradi, a on će se odmah odazvati.

### Pisanje prvog programa

Da biste započeli interaktivnu sesiju, u meniju Start izaberite opciju All Programs, Python 3.1, IDLE (Python GUI). Trebalo bi da na ekranu ugledate nešto nalik onome na slici 1-4.

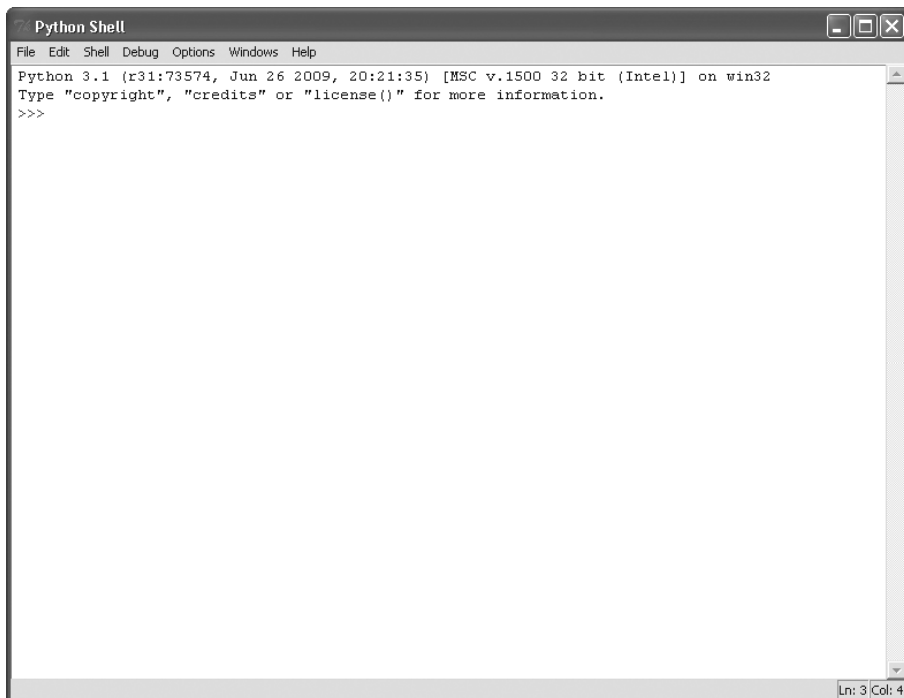
Taj prozor zove se Python Shell. U isti red, iza komandnog odzivnika (`>>>`), upišite: `print("Game Over")`, pa pritisnite taster Enter. Interpreter se odaziva tako što na ekranu prikazuje sledeće:

```
Game Over
```

Čestitamo! Upravo ste napisali svoj prvi Python program! Sada ste pravi programmer (koji treba da nauči još ponešto, ali to važi za sve nas).

### Upotreba funkcije print

Pogledajte red teksta koji ste upisali, `print("Game Over")`. Obratite pažnju na to koliko je lako razumljiv sam po sebi. Čak i bez ikakvog znanja o programiranju, verovatno biste pogodili šta radi. To je suština Pythona. Sažet je i jasan sam po sebi. To ćete još više ceniti kada naučite da radite složenije stvari na tom jeziku.



Slika 1-4: Python u interaktivnoj sesiji, dok čeka vašu komandu.

Funkcija `print()` prikazuje na ekranu tekst koji zadate u navodnicima između njenog para zagrada. Ako između zagrada ne zadate ništa, ispisaće prazan red.



Python pravi razliku između malih i velikih slova, a imena funkcija pišu se malim slovima. Tako da je `print("Game Over")` ispravno, ali `Print("Game Over")` i `PRINT("Game Over")` nije.

## Učenje žargona

Pošto ste sada postali programer, moraćete da „prosipate“ unaokolo one čudne izraze koje samo programeri razumeju. **Funkcija** je nalik mini programu koji se pokreće i obavlja određeni posao. Posao funkcije `print()` je da na ekranu prikaže zadatu vrednost (ili niz vrednosti). Funkciju pokrećete, to jest, **pozivate** navodeći njeno ime, čemu sledi par zagrada. Tačno to ste uradili u interaktivnom režimu kada ste otkucali `print("Game Over")`. Ponekad možete zadati, tj. **proslediti** funkciji ulazne vrednosti s kojima će ona raditi. Te vrednosti, koje se zovu **argumenti**, treba navesti između zagrada (iza imena funkcije). U primeru vašeg prvog programa, funkciji `print()` prosledili ste argument "Game Over", koji je funkcija upotrebila da bi ispislala poruku Game Over.



Funkcije u Pythonu takođe **vraćaju** podatke, tj. prosleđuju ih nazad u deo programa koji je pozvao funkciju. Te vrednosti se zovu **povratne vrednosti**. Više o povratnim vrednostima saznaćete u poglavlju 2.

U ovom konkretnom primeru, možete biti još precizniji ako kažete da je vrednost "Game Over" koju ste prosledili funkciji `print()` – **znakovni niz** (engl. *string*). To samo znači da je u pitanju niz pojedinačnih znakova, kao što su oni na vašoj tastaturi. Možda vam „znakovni niz“ izgleda kao neobično ime – izrazi „tekst“ ili „reči“ bili bi možda jasniji – ali, ideja se zasniva na činjenici da se tekst sastoji od grupe, tj. niza pojedinačnih znakova. Tehnički gledano, "Game Over" je **znakovni literal** zato što je to doslovno niz znakova koji čine reči.

Red teksta koji ste upisali u interpreter takođe se smatra  **naredbom** (engl. *statement*). U govornom jeziku naredba predstavlja jednu celovitu misao. U Pythonu, naredba je celovit opis nečega što treba uraditi. Svaki program sadrži određen broj naredaba.

I najzad, pošto ste sada programer, možete pričati drugima kako ste napisali nešto Pythonovog **programskog koda**. Programski kôd su programske naredbe. Postoji i izveden glagol **kodirati** koji opisuje radnju programiranja. Na primer, možete reći da ste celu noć proveli grickajući čips, pijuci sokove i **kodirajući** kao sumanut.

## Greške u kodu

Računari sve razumeju doslovno. Ako u imenu funkcije pogrešite samo jedno slovo, računar neće imati ni najmanjeg pojma šta ste imali na umu. Recimo, ako na interaktivnom odzivniku otkucate `print("Game Over")`, interpreter će se odzvati nečim nalik na ovo:

```
Traceback (most recent call last):
  File "<pyshell#0>", line 1, in <module>
    print("Game Over")
NameError: name 'print' is not defined
```

Prevedeno na običan jezik, interpreter samo kaže „Molim?!“ Najvažniji red u poruci o grešci je `NameError: name 'print' is not defined`. Ta poruka znači da interpreter ne prepoznaje pojam `print`. Kao ljudsko biće, vi možete zanemariti svoju grešku u kucanju i razumeti šta sam imao na umu. Računari nisu toliko tolerantni. Srećom, programske greške (engl. *bugs*), kao što je ova, lako se otklanjaju ispravljanjem greške u kucanju.

## Isticanje delova sintakse

Verovatno ste zapazili da se reči prikazuju na ekranu različitim bojama (ne i u knjizi, razume se). Upotreba boja na taj način, što se zove **isticanje delova sintakse**, pomaže vam da brzo shvatite šta ste otkucali tako što ga vizuelno razvrstate. U tom šarenilu boja postoji sistem. Specijalne reči koje su sastavni deo jezika Python, kao što je `print`, prikazuju se ljubičastom bojom. Znakovne vrednosti, kao što je "Game Over", obojene su u zeleno. A rezultati – ono što interpreter prikazuje kao posledicu onoga što ste otkucali – obojeni su u plavo. Kada budete pisali opsežnije programe, taj sistem bojenja pomoći će vam da jednim pogledom pregledate svoj kôd i lakše uočite greške u njemu.

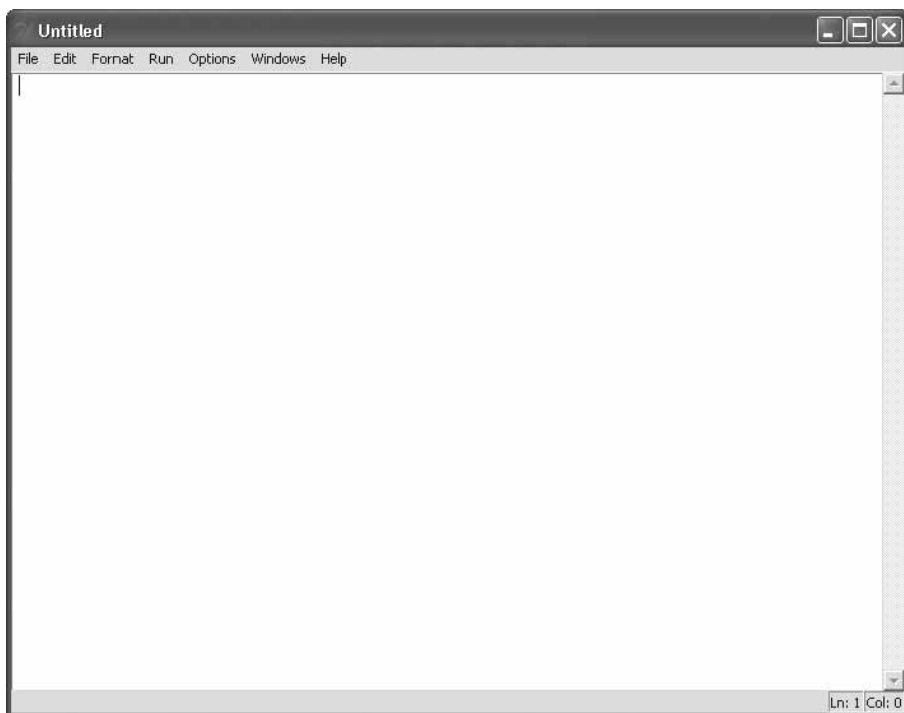


## Programiranje u skript režimu

Kada radite u interaktivnom režimu, odziv je trenutan. To je dobro jer možete odmah videti rezultate. Ali taj režim nije pogodan za pisanje programa koje možete da sačuvate i izvršavate kasnije. Srećom, Pythonovo okruženje IDLE takođe stavlja na raspolaganje i skript režim, u kojem možete pisati, menjati, učitati i snimati svoje programe na disk. To je kao da za svoj kôd imate alatku nalik programu za obradu teksta. U stvari, možete obavljati poslove koji su uobičajeni u programima za obradu teksta, kao što su pronalaženje zadatih pojmova i njihovu zamenu drugim, ili kopiranje i umetanje blokova teksta.

### Pisanje prvog programa (ponovo)

Prozor za rad u skript režimu možete otvoriti iz interaktivnog prozora u kojem ste dosad radili. Izaberite meni File, a zatim opciju New Window. Pojaviće se nov prozor nalik onom sa slike 1-5.



**Slika 1-5:** Prazno platno vas čeka. Python je spreman i čeka da napišete program u skript režimu.

U tom novom prozoru za rad u skript režimu, upišite pri nt ("Game Over") i pritisnite Enter. Ništa se ne događa! To je zato što ste u skript režimu, gde samo pišete listu naredaba koje će računar izvršiti kasnije. Pošto program snimate na disk, možete ga izvršiti.

## Snimanje programa na disk i njegovo izvršavanje

Da biste program snimili na disk, izaberite File, Save As. Svom primerku sam nadenulo ime `game_over.py`. Da bih kasnije lakše dolazio do njega, snimio sam ga na radnu površinu računara.



Programne obavezno snimajte s nastavkom (oznakom tipa) `.py`. To omogućava da razne aplikacije, među kojima i IDLE, prepoznaju te datoteke kao Pythonove programe.

Da bih pokrenuo izvršavanje programa Game Over, treba samo da izaberem Run, Run Module. U interaktivnom prozoru se prikazuju rezultati mog programa. Pogledajte izgled moje radne površine na slici 1-6.

```

Python Shell
File Edit Shell Debug Options Windows Help
Python 3.1 (r31:73574, Jun 26 2009, 20:21:35) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> print("Game Over:")
Game Over
>>> ----- RESTART -----
>>>
Game Over
>>>
Ln: 8 | Col: 4

```

Slika 1-6: Rezultati izvršavanja programa Game Over u okruženju IDLE.

Obratite pažnju na to da interaktivni prozor sadrži stari tekst unet ranije. I dalje sadrži naredbu koju sam upisao dok sam radio u interaktivnom režimu, `print("Game Over")`, kao i njen rezultat, poruku `Game Over`. Ispod toga je reč `RESTART`, a ispod nje su rezultati programa koji je izvršen u skript režimu: `Game Over`.

Da biste pokrenuli program iz okruženja IDLE, morate ga prvo snimiti negde na disk.

Interaktivni režim je odličan za brzo isprobavanje jednostavnijih ideja. Skript režim je savršen za pisanje programa koje kasnije izvršavate. Kombinovanje oba režima odličan je način za pisanje programskog koda.

Uprkos tome što mi je za pisanje programa dovoljan samo skript režim, uvek držim otvoren i interaktivni prozor. Dok pišem programe u skript režimu, prelazim u interaktivni prozor kako bih isprobao neku ideju ili proverio da li ispravno pozivam odedenu funkciju.

Skript prozor je mesto gde pripremam konačni proizvod. Interaktivni prozor je nalik papirnom bloku s kojim eksperimentišem. Kada kombinujem oba prozora, brže pišem programe i oni su bolji.

## Povratak na program Game Over

Izvršavanje verzije programa Game Over dosad ste pokretali iz okruženja IDLE. Dok razvijate dati program, njegovo pokretanje iz okruženja IDLE dobar je način rada. Ali, siguran sam da želite da se vaši završeni proizvodi pokreću na isti način kao i svi drugi programi na vašem računaru. Želite da korisnik samo dvaput mišem pritisne ikonicu vašeg programa kako bi ga pokrenuo.

Ako biste na taj način pokušali da pokrenete verziju programa Game Over koju sam dosad pokazao, videli biste samo da se otvara prozor i da se jednako brzo zatvara. Verovatno biste pomislili da se ništa nije dogodilo. Nešto bi se ipak dogodilo, samo što bi to bilo prebrzo da biste išta zapazili. Program bi se pokrenuo, prikazao bi se tekst `Game Over` i program bi se završio, a sve to u deliću sekunde. Programu je potreban način da zadrži otvoren svoj prozor konzole.

U izmenjenoj verziji programa Game Over, poslednjeg projekta u ovom poglavlju, zadržava se otvoren prozor programa da bi korisnik video poruku. Pošto prikaže natpis `Game Over`, program prikazuje i poruku `Press the enter key to exit` (pritisnite taster `Enter` da biste završili). Pošto korisnik pritisne taster `Enter`, program se završava, a prozor konzole nestaje.

Vodiću vas kroz kôd odeljak po odeljak. Program možete učitati s prateće Web lokacije ([www.courseptr.com/downloads](http://www.courseptr.com/downloads)), iz direktorijuma Chapter 1; ime datoteke je `game_over.py`. Još bolje je da sami prepisete program i izvršite ga.



Pod operativnim sistemom Windows, možete direktno otvoriti Pythonov program u okruženju IDLE ako desnim tasterom miša pritisnete ikonicu datoteke i izaberete opciju `Edit with IDLE`.

## Komentari u kodu

Prva dva reda programa izgledaju ovako:

```
# Game Over  
# Demonstrates the print function
```

Ti redovi nisu naredbe koje računar treba da izvrši. U stvari, računar ih potpuno zanemaruje. Te napomene, nazvane *komentari* (engl. *comments*) namenjene su ljudima. Komentari objašnjavaju programski kôd rečima govornog jezika. Izuzetno su korisni drugim programerima i pomažu im da razumeju vaš kôd. Ali, korisni su i vama jer vas podsećaju kako ste uradili nešto što nije jasno na prvi pogled.

Komentar umećete pomoću simbola „taraba“, `#`. Sve što sledi tom simbolu (osim kada se on nalazi unutar znakovnog niza) u preostalom delu reda, smatra se komentarom. Računar zanemaruje kometanatare. Obratite pažnju na to da se u okruženju komentari prikazuju crvenom bojom kako bi se lakše uočili.

Dobra ideja je da svaki program započnete s nekoliko redova komentara, kao što sam ja uradio u navedenom primeru. Korisno je da navedete naslov i svrhu programa. Mada ja to u ovom primeru nisam uradio, trebalo bi da navedete i ime programera i datum kada je program napisan.

Možda sada mislite: „Šta će mi uopšte komentari? Pošto sam ja napisao program, znam šta on radi“. To će možda biti tačno još mesec dana pošto završite pisanje koda, ali iskusni programeri znaju da posle nekoliko meseci bavljenja sasvim drugim stvarima, vaše prvobitne namere možda vam neće biti tako jasne. Ako želite da izmenite neki stari program, nekoliko dobro osmišljenih komentara mogu vam znatno olakšati život.

---

## U STVARNOM SVETU

Komentari su još korisniji drugom programeru koji treba da izmeni vaš program. Ta vrsta situacije česta je u svetu profesionalnog programiranja. U stvari, procenjuje se da programer troši veći deo svog vremena i truda na održavanje postojećeg koda nego na pisanje novog. Nije neuobičajeno da programer dobije zadatak da izmeni program koji je neko drugi napisao – a postoji verovatnoća da prvobitni programer ne bude na raspolaganju kako bi odgovarao na pitanja. Iz tog razloga, dobri komentari su izuzetno korisni.

---

## Prazni redovi

Tehnički gledano, sledeći red u programu je prazan. Računar uglavnom zanemaruje prazne redove; oni su takođe namenjeni ljudima koji čitaju kôd. Prazni redovi mogu da olakšaju razumevanje programa. Redove koji čine blokove povezanog koda najčešće razdvajam praznim redom. U ovom programu, praznim redom sam razdvojio komentare od poziva funkcije `print`.

## Prikazivanje znakovnog niza na ekranu

Trebalo bi da prepoznate sledeći red u programu:

```
print("Game Over")
```

To je vaš stari prijatelj, funkcija `print`. Taj red, isto kao i u interaktivnom režimu, prikazuje natpis `Game Over`.

## Čekanje na odziv korisnika

Poslednji red programa:

```
input("\n\nPress the enter key to exit.")
```

prikazuje odzivnik, `Press the enter key to exit`, i čeka da korisnik pritisne taster `Enter`. Pošto korisnik pritisne taj taster, program se završava. To je zgodan trik koji omogućava da prozor konzole ostane otvoren dok korisnik ne završi rad sa aplikacijom.

---

Sada bi otprilike bilo vreme da objasnim šta se događa u tom redu. Ali, držaću vas u iščekivanju. Izvinjavam se. Moraćete da sačekate do sledećeg poglavlja kako biste u pravoj meri cenili taj jedan jedini red.

## Sažetak

U ovom poglavlju prešli ste prilično dug put. Saznali ste nešto o Pythonu i njegovim jačim stranama. Instalirali ste jezik na svoj računar i malo ga isprobali. Naučili ste da radite u Pythonovom interaktivnom režimu kako biste izvršili pojedinačne programske naredbe. Videli ste kako se koristi skript režim da biste pisali, menjali, snimali na disk i izvršavali duže programe. Naučili ste da prikazujete tekst na ekranu i da sačekate da korisnik zatvori konzolski prozor programa. Postavili ste temelj za svoju avanturu zvanu programiranje na jeziku Python.

---

## ZADACI

1. Namerno izazovite grešku tako što ćete otkucati naziv svoje omiljene vrste sladoleda u interaktivnom režimu. Zatim, da biste se iskupili, napišite naredbu koja ispisuje naziv vaše omiljene vrste sladoleda.
  2. Napišite i snimite na disk program koji ispisuje vaše ime, a zatim čeka da korisnik pritisne taster Enter pre nego što se završi. Zatim pokrenite taj program tako što dvaput mišem pritisnete njegovu ikonicu.
  3. Napišite program koji ispisuje vaš omiljeni citat. U sledećem redu treba da prikaže ime autora citata.
-

