Autor: Renato Majer

Car GAMe

Tehnička dokumentacija

Sadržaj

[UVOD 2](#_Toc39786457)

[AUTOR 2](#_Toc39786458)

[IDEJA PROJEKTA 2](#_Toc39786459)

[OPIS IGRE 2](#_Toc39786460)

[DETALJAN OPIS RADA 3](#_Toc39786461)

[PROGRAMSKI JEZIK 3](#_Toc39786462)

[TIJEK RADA 3](#_Toc39786463)

[STVARANJE PROZORA 4](#_Toc39786464)

[POSTAVLJANJE GLAVNE PETLJE 4](#_Toc39786465)

[PRIKAZIVANJE OBJEKATA I IGRAČA 5](#_Toc39786466)

[OSNOVNE MEHANIKE I KRETANJE OBJEKATA I IGRAČA 6](#_Toc39786467)

[INTERAKCIJA OBJEKATA 6](#_Toc39786468)

[SCORE 7](#_Toc39786469)

[PRIKAZIVANJE TEKSTA I UPDATE 7](#_Toc39786470)

# UVOD

## AUTOR

Moje ime je Renato Majer, učenik sam 4 razreda Gimnazije Andrije Mohorovičića Rijeka. Pohađajući matematički smjer, imao sam priliku na mnogima satima informatike dobro se upoznati sa Pythonom i još nekolicinom programskih jezika te steći osnove programiranja. Dalje gradeći svoje znanje vezano uz programiranje, postigli smo određenu razinu, na kojoj već i sami možemo stvarati vlastite projekte, te smo se u tome i okusili.

## IDEJA PROJEKTA

Na satu informatike dobili smo zadatak stvaranja vlastite računalne igre. Samim time, morali smo odabrati određeni programski jezik i određenu temu na koju ćemo stvarati. Kako smo se u protekle četiri godine najviše koristili Pythonom, odlučio sam se upravo za uporabu istog i u ovom projektu. Samu temu računalne igre dobio sam na način da sam se prisjetio „davnih“ osnovnoškolskih dana i sati provedenih igrajući igrice na računalu. Kako je tada moja najdraža tema video igrica bila usko povezana sa automobilima, odlučio sam se tu istu temu iskoristiti i ovdje.

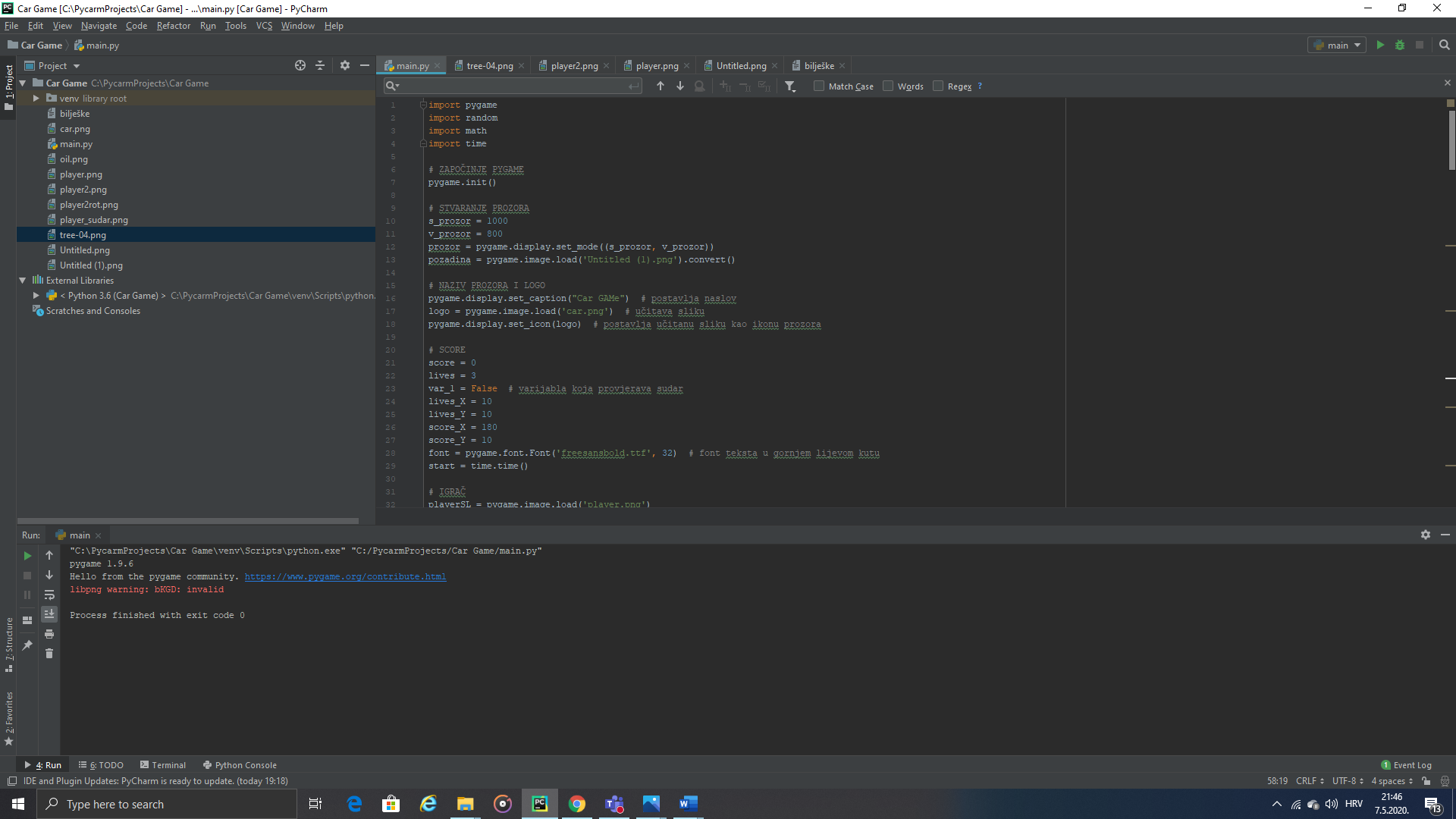
## OPIS IGRE

Glavna je ideja igrice putovanje cestom u crvenom automobilu, dok pritom pokušavate izbjeći sve prepreke na koje putem nailazite. Prepreke se u igrici pojavljuju u obliku automobila koji nailaze iz suprotnog smjera i mrlja ulja na kolniku. Tijekom samog igranja povećava se razina „teškoće“ igranja, te se tako prepreke na kolniku sve brže i brže kreću prema samom igraču. Igrač pri pokretanju igrice posjeduje tri života. Također, od pokretanja igrice, igraču se bilježi postignuće, odnosno „score“. Pri svakom nabasanju na prepreku, igraču se oduzima po jedan život, te se naposlijetku, u trenutku kada igrač ostaje bez života, igrica prekida i ispisuje njegov konačni score. Score se u igrici računa pomoću sekunda igranja. Odnosno, svaka sekunda koju provedete igrajući, predstavlja po jedan poen. Cilj je igrice postići što veći broj poena.

# DETALJAN OPIS RADA

## PROGRAMSKI JEZIK

Pri samom stvaranju programa, kao što je već prije napomenuto, korišten je programski jezik Python. Python je korišten upravo zbog autorovog dobrog poznavanja tog programskog jezika, te zbog njegove jednostavnosti, lakoće korištenja i baratanja istim. Igrica je pisana pomoću modula Pygame. Također, pri stvaranju projekta korišteno je još nekoliko modula. Modul Random korišten je za nasumično dobivanje brojeva koje određuju koordinate pojavljivanja prepreka. Modul Math korišten je pri nekim matematičkim operacijama, dok je modul Time korišten za mjerenje trajanja same igre, te se na taj način određivao score.

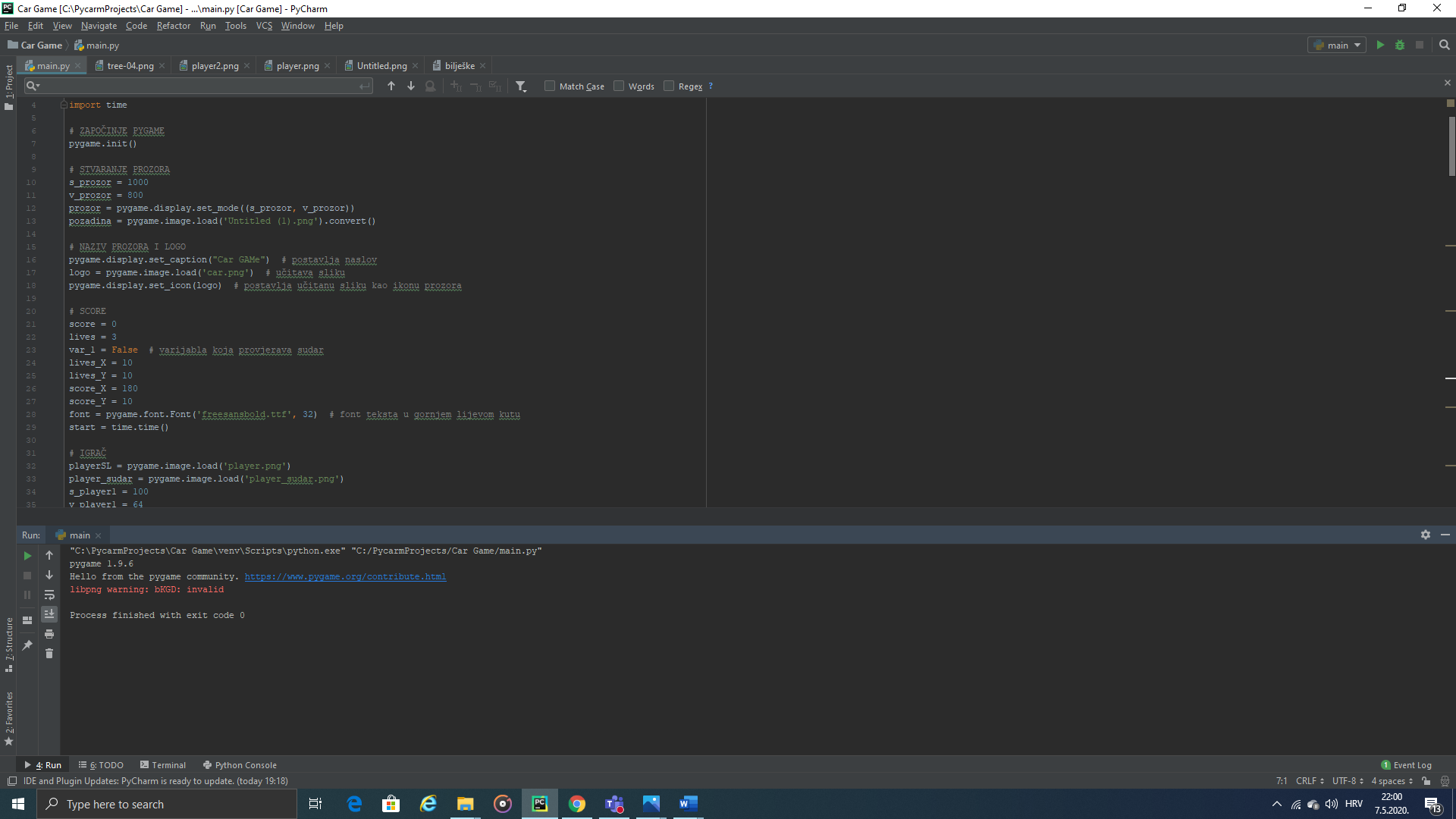


## TIJEK RADA

Prvi korak u početku rada nakon određivanja teme i smišljanja koncepta igrice bilo je pisanje samog koda. Prije početka pisanja koda, bilo je potrebno upoznavanje i shvaćanje osnovnih funkcija i metoda modula Pygame kako bi samo stvaranje igrice teklo glatko. Nakon toga, implementacijom znanja stečenog u proučavanju Pygame-a, nastao je prvi kod sa osnovnim funkcijama i prvotnim „screenom“. Dalje su stvarane i ostale složenije funkcije koje određuju složenije procese, a čiji će rad biti opisan u nastavku. Vrijeme potrebno za stvaranje i finalizaciju samog projekta bilo je 35 dana.

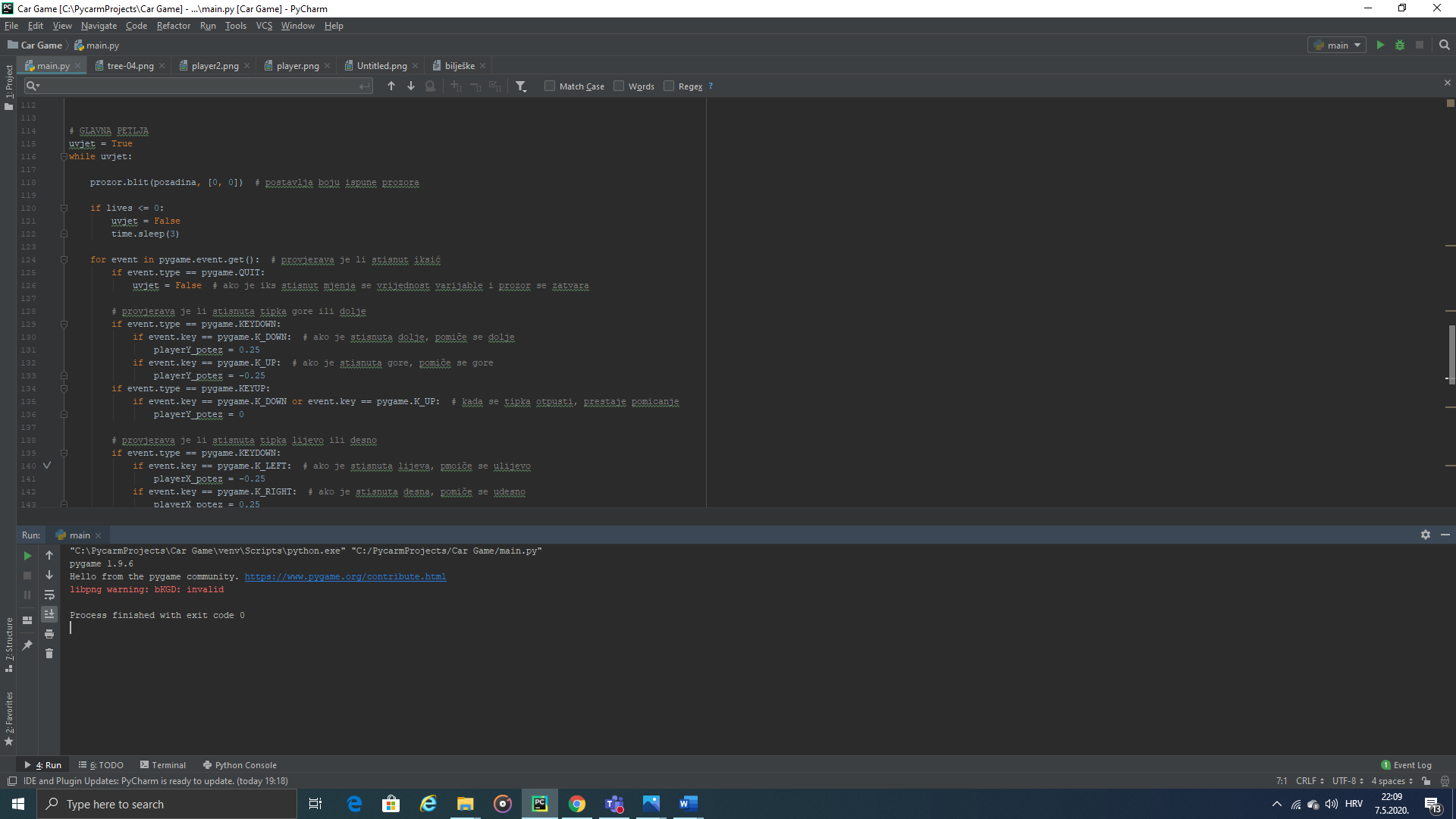
## STVARANJE PROZORA

U programu je na samom početku pozvana funkcija pygame.init() koja započinje inicijalizira Pygame. Nakon toga, određene su dimenzije prozora, te je na prozor dodana pozadina. Na zaglavlje prozora dodan je naziv igrice „Car GAMe“ i sam logo igrice.



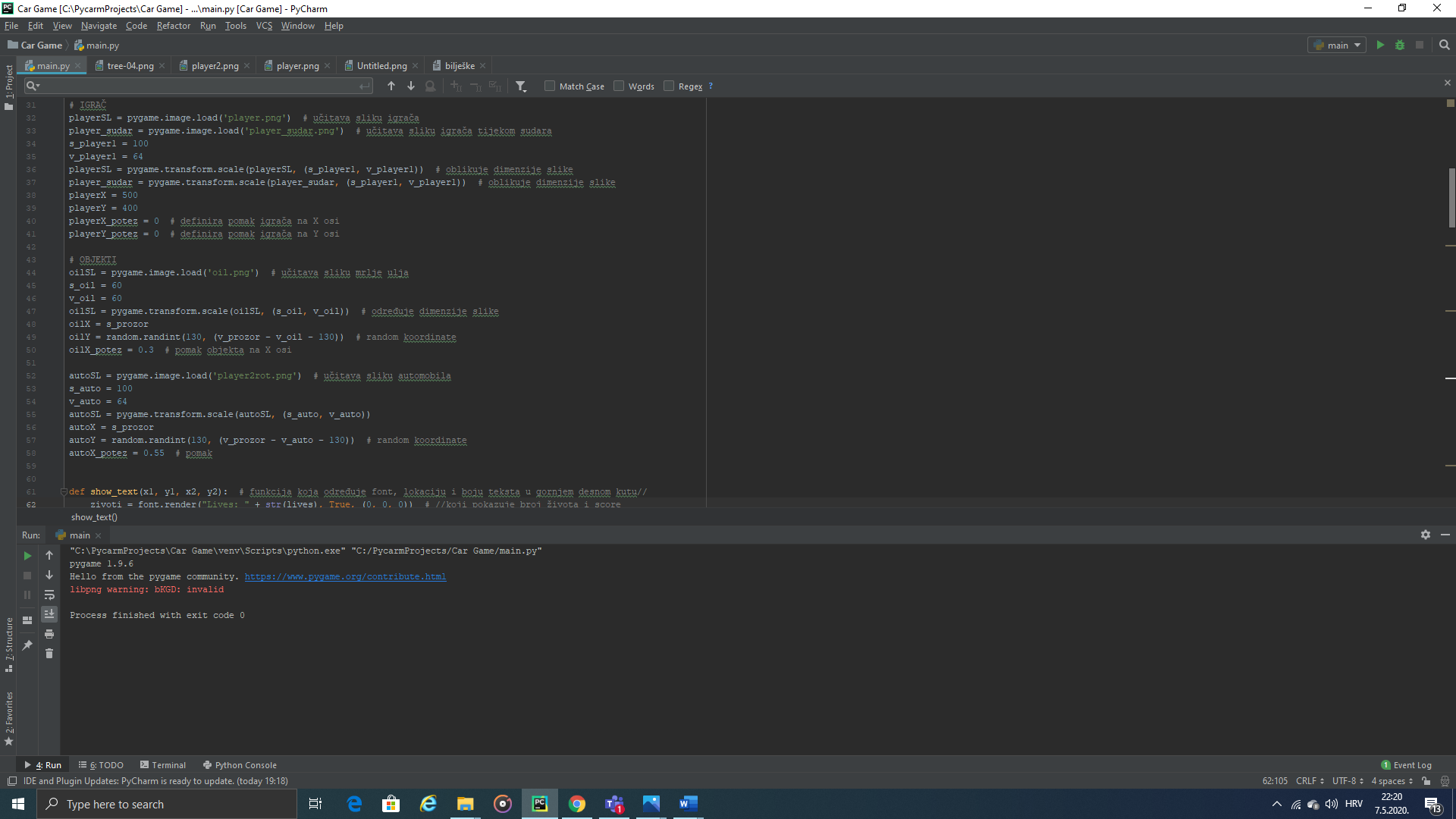
## POSTAVLJANJE GLAVNE PETLJE

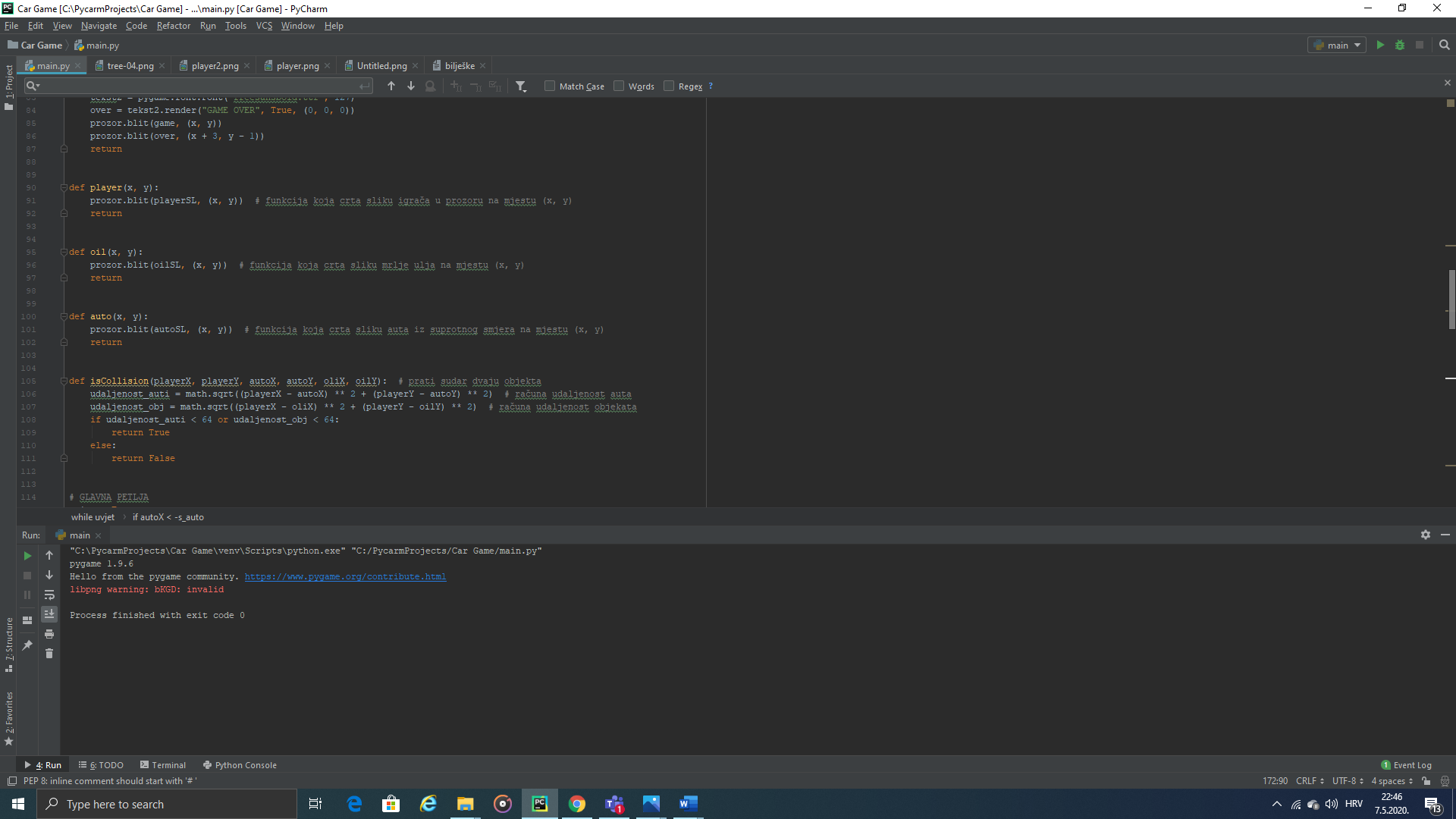
Nakon što je stvoren glavni prozor na kojem će se odvijati sva radnja igrice, bilo je potrebno stvoriti glavnu petlju koja će određivati samo zatvaranje glavnog prozora i završetak igrice. Za tako nešto korištena je „while“ petlja koja je uvjetovana jednom varijablom čija je vrijednost postavljena na „True“. U glavnoj se petlji nalazi provjera događaja koja kontrolira je li pritisnut iksić u gornjem desnom kutu prozora, te ako je, mijenja vrijednost varijable, te se glavna petlja prekida i prozor se zatvara. Također, na početku je glavne petlje dodana i varijabla koja kontrolira broj života, te ako se broj života izjednači sa nulom, igrica se također zatvara.



## PRIKAZIVANJE OBJEKATA I IGRAČA

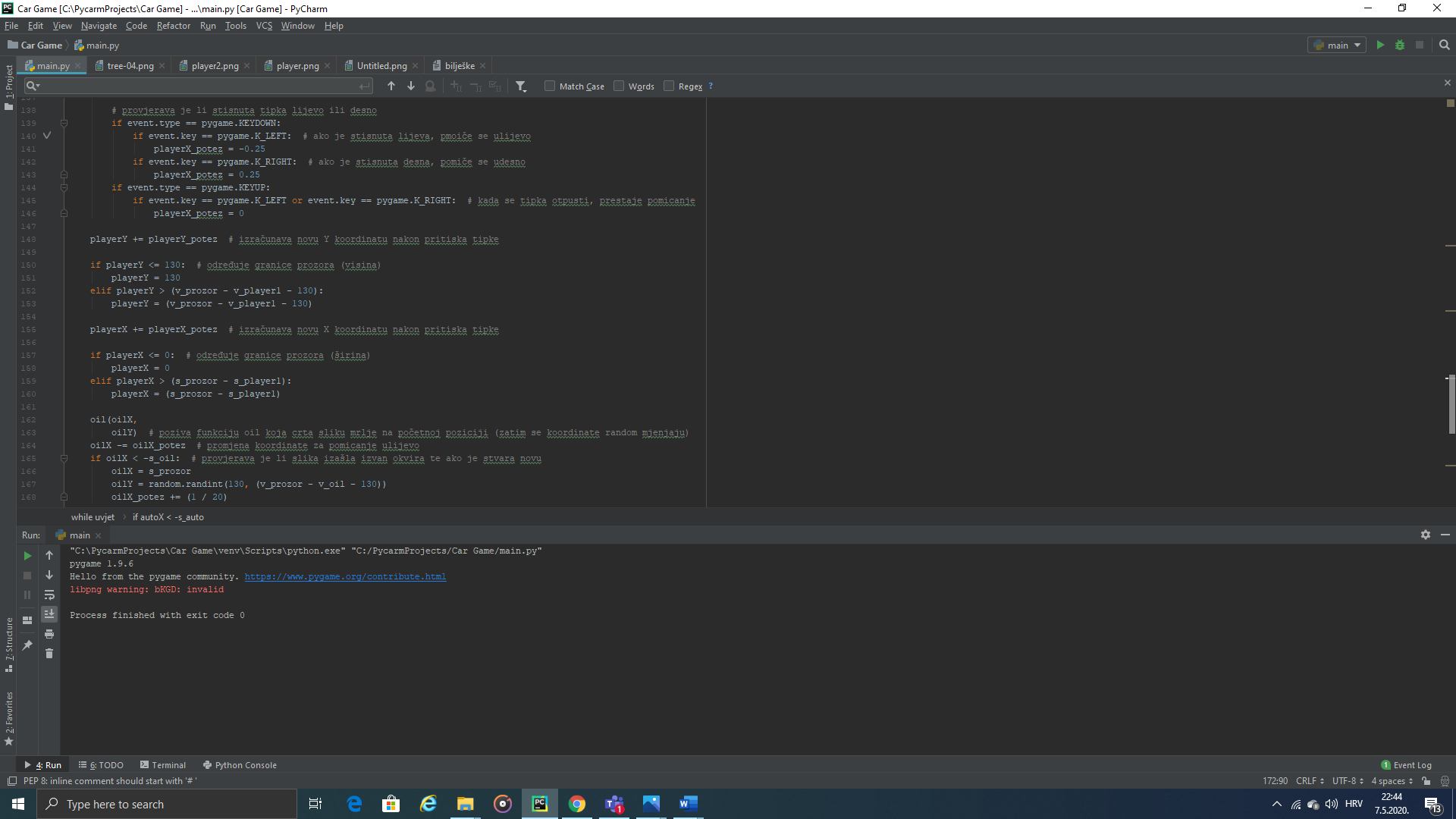
U program su učitane slike objekata i igrača, te su varijablama određene njihove dimenzije. Nakon toga, slike su uz pomoć nekih funkcija modula Pygame prilagođene dimenzijama potrebnima za samo korištenje u igrici. Kasnije je učitana i slika igrača u trenutku sudara, te su definirane varijable vezane uz pomak samih objekata i igrača. Također su definirane i „random“ koordinate objekata. Izvan glavne ptelje definirane su i funkcije čija je zadaća prikazivanje slika na glavnom screenu. Te su funkcije kasnije pozvane u glavnoj petlji.





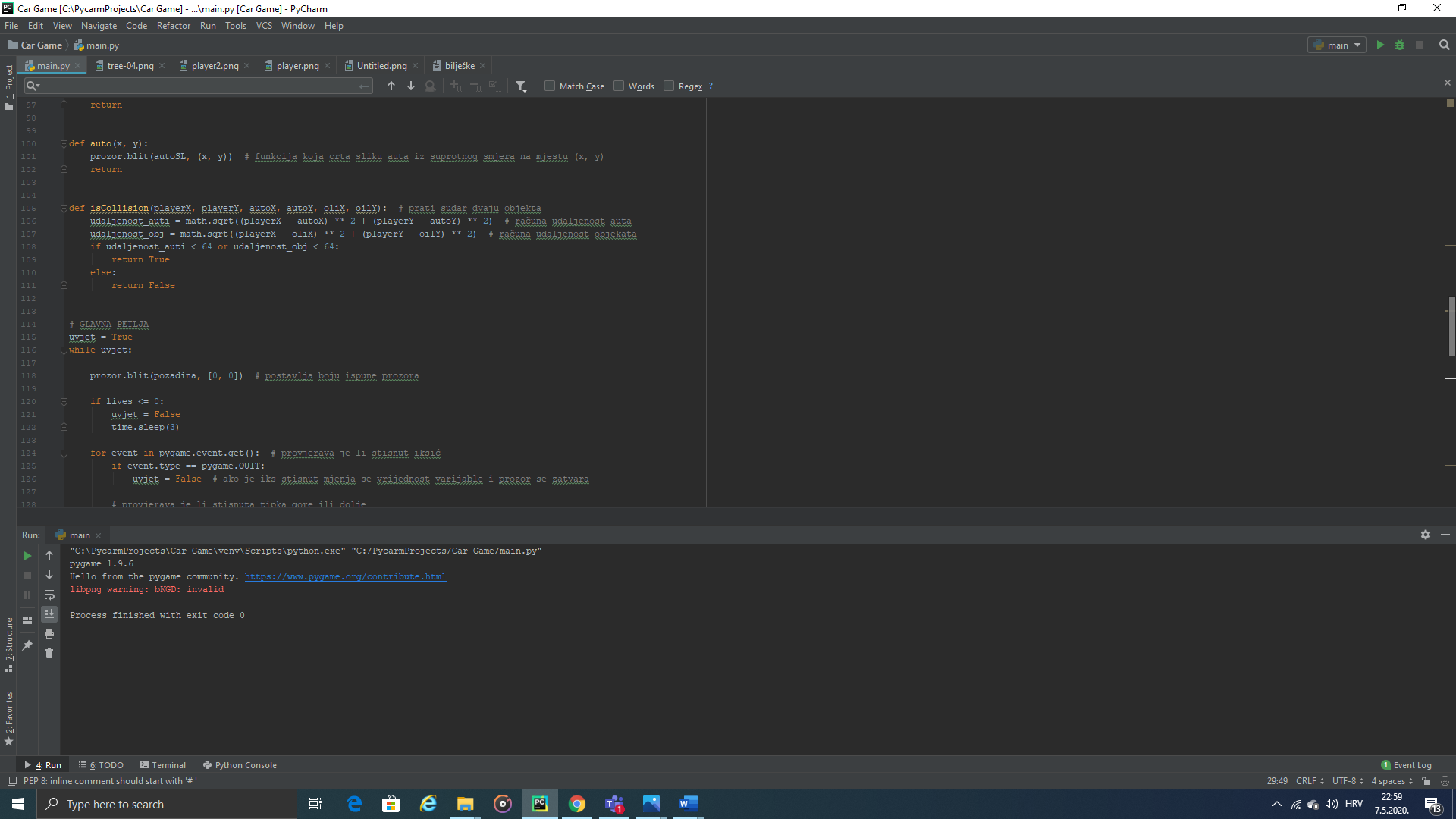
## OSNOVNE MEHANIKE I KRETANJE OBJEKATA I IGRAČA

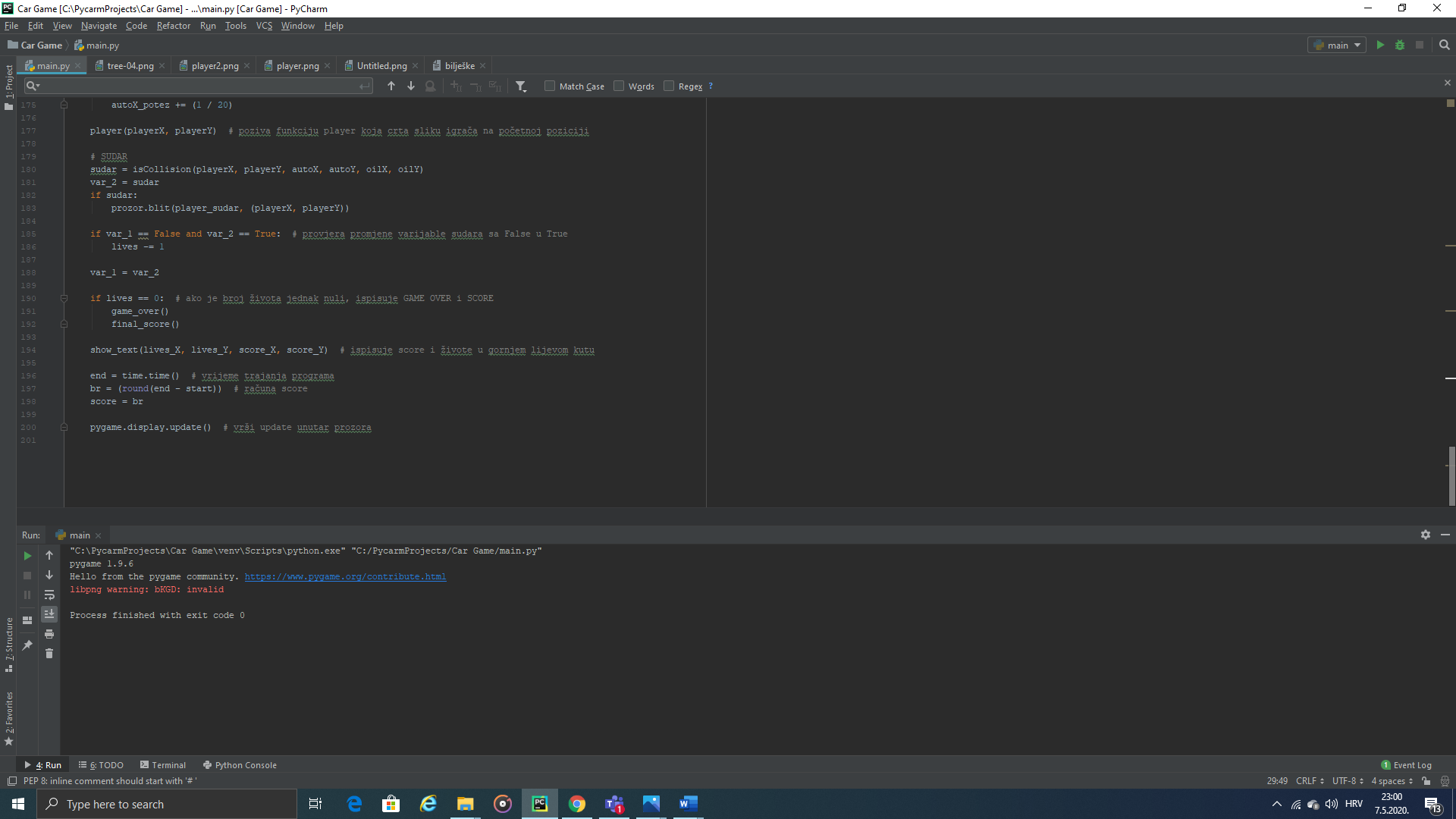
Korištenjem funkcija modula Pygame, provjeravano je je li stisnuta koja od četiri tipke te je tim tipkama pridruženo određeno kretanje samog igrača. Također, kako se objekt ili igrač ne bi kretali do beskonačnosti, određene su granice njihova kretanja koje se podudaraju sa granicama samog prozora. Kada objekt pređe granice prozora, on nestaje i pojavljuje se novi objekt sa random koordinatama.



## INTERAKCIJA OBJEKATA

Sama interakcija objekata definirana je na način da je izračunata udaljenost dvaju objekta (principom matematičkog računanja udaljenosti dviju točaka u ravnini), te se kolizija pojavljuje ako je njihova udaljenost manja od same širine ili dužine objekata. Prilikom sudara, umanjuje se varijabla koja određuje broj života. Nadalje, kada se ustanovi da je došlo do interakcije objekata, slika se igrača mijenja te je on prikazan u svjetlijoj ili „light“ verziji. Također, dvije su varijable specijalno definirane i prikazuju stanja „True“ i „False“ te one definiraju trenutak početka i završetka same interakcije objekata kako bi omogućile lakše računanje i izvođenje ostaliz funkcija i operacija. Na fotografijama ispod, prikazana je funkcija koja definira interakciju, te sama implementacija funkcije, te računanje života, itd.



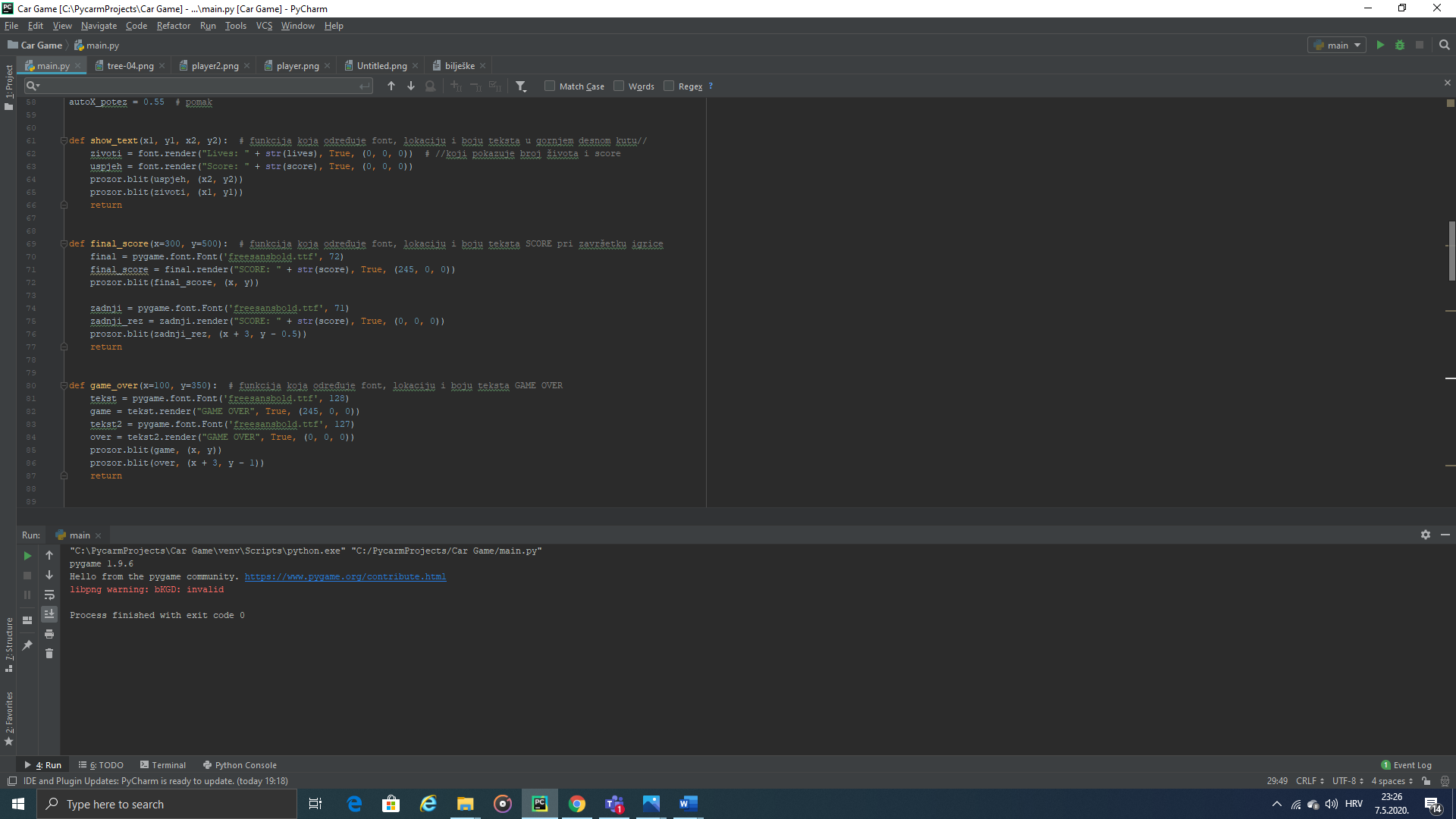


## SCORE

Score se računa od početka igranja pa sve do kraja ili završetka igrice. Pozivanjem funkcije time iz istoimenog modula započinje se na samom početku mjeriti vrijeme. Isto tako pri svakom se izvršavanju glavne petlje poziva još jedna takva funkcija te se konačna vrijednost vremena oduzima od početne. Broj sekundi od početka do kraja izvršavanja glavne petlje ili od početka do kraja igranja igrice predstavlja poene, dok poeni čine score. U suštini, igrač koji dulje igra, ima veći score. Kako se brzina nadolazećih objekata prilikom svakog novog ulaska u glavnu petlju povećava, score se sve teže prikuplja zbog povećanja težine igranja, odnosno težinskog levela same igrice.

## PRIKAZIVANJE TEKSTA I UPDATE

Pomoću posebnih funkcija modula Pxgame, definirano je prikazivanje teksta i ispisa vrijeble score i lives (životi). Tekst se ispisuje u gornjem lijevom kutu te se prilikom svakog ulaska u glavnu petlju vrši update samog teksta, zbog mijenjanja score-a i broja života. Tekstu je definiran određeni font, veličina i boja. Također, varijablama koje u sebi sadrže broj koordinata, sam je tekst precizno pozicioniran na željenu poziciju. Kako bi se tekst i cijeli screen u potpunosti ažurirali i kako bi se prikazale sve moguće funkcije i grafički prikazače, prilikom svakog novog ulaska u glavnu petlju, pri kraju je petlje potrebno pozvati funkciju pygame.display.update() koja vrši update unutar prozora.  
Nadalje, prilikom gubljenja svih života, igrica se „freeza“ te se velikim dimenzijama preko cijelog screena ispisuje „GAME OVER“ i „SCORE“ te njegova vrijednost. To je ispisivanje finalnog teksta definirano posebnim funkcijama koje ispisuju tekst dva puta i u različitim bojama, kako bi se stekao dojam trodimenzionalnosti i pružio pozitivan estetski dojam. Nakon tog ispisivanja, program se pomoću modula time zaleđuje na 3 sekunde te se tako i glavni screen zaleđuje. Igrač može sa screena iščitati svoj score i uvidjeti kako je igrica završila te postati svjestan svog rezultata. Nakon što vrijeme od 3 sekunde istekne, igrica se gasi i glavni se screen zatvara.



# TEHNIČKE INFORMACIJE

Pri stvaranju samog projekta korištene su brojne tehnologije i resursi kako bi realizacija projekta bila moguća. Crpilo se znanje i informacije iz nekoliko izvora kako bi se shvatilo određene procese i funkcije. Korišteni su alati popout „Photoshopa“ za stvaranje vlastitih slika koje su korištene u igrici te „Screencast O'Matic“ kao alat za stvaranje screencasta.

Sistemska konfiguracija računala za pokretanje ovog projekta, odnosno otvaranje ove igrice je gotovo zanemariva, zbog toga što se igrica može pokrenuti vrlo lako i ne zahtjeva složene operacije za njeno izvođenje te ne zauzima mnogo memorije.

Ovaj mi je projekt pružio dublji uvid u programiranje te stvaranje grafičkih sučelja te potaknuo na daljnje istraživanje i razvoj.