

Uputstva za anotacija komentara iz programskog koda

Komentari u programskom sadrže bogate informacije o programskom kodu koji opisuju. To može da bude opis funkcionalnosti, navođenje autora, licence, način korišćenja i sl. Cilj ovog zadatka je klasifikacija zadatih komentara u jednu od osam definisanih kategorija. Komentari su pisani na srpskom ili na engleskom jeziku i dolaze iz koda pisanog u nekom od sledećih jezika: C, C++, Java, Python, JavaScript, TypeScript, PHP, SQL, C#

Komentari mogu biti funkcionalni i nefunkcionalni. Funkcionalni komentari su komentari koji opisuju funkcionalnost odgovarajućeg koda. Ti opisi mogu biti kratki ili da se sadrže nekoliko redova teksta. Obično se koriste da opišu namenu i ponašanje koda, ili razlog zašto je nešto implementirano na određeni način. Uglavnom odgovaraju na pitanja *Šta?*, *Zašto?* i *Kako?* Na osnovu tipa koda koji opisuju, ovi komentari pripadaju jednoj od sledeće tri kategorije:

1. *Functional-Inline* – komentari u liniji koji opisuju funkcionalnost određene linije koda. To na primer može biti opis promenljive ili nekog izraza unutar tela metode ili funkcije. Na primer:

```
define("VALUTA",118); //ovde se dodeljuje const vrednost promenljivoj valuta
```

2. *Functional-Method* – komentari koji opisuju funkcionalnost određene funkcije/metode. Obično se nalaze iznad početka deklaracije ili definicije metode. Često pored opisa funkcionalnosti sadrže i opis argumenata i povratne vrednosti. Na primer:

```
/**
 * U ovoj funkciji bice izvršena provera podataka iz baze
 * i vracena vrednost statusa
 * @param Place $mesto Lokacija za koju trazimo podatke
 * @param integer $ptt Postanski broj mesta za koje trazimo podatke
 *
 * @throws Some_Exception_Class Ako ce nesto biti kao izuzetak
 * @return Status
 */
```

3. *Functional-Module* – komentari koji opisuju funkcionalnost određenog modula kao što je na primer klasa ili interfejs. Kod programskih jezika koji ne poznaju objektno-orjentisane koncepte ovakvi komentari se odnose na ceo fajl. Na primer:

```
/**
 * Klasa DataBaseHandler implementira sve metode koje su potrebne za
 * komunikaciju sa specificiranom bazom podataka.
 *
 */
```

Nasuprot funkcionalnim komentarima postoje nefunkcionalni komentari koji pokrivaju sve kategorije komentara koji nisu funkcionalni. Oni mogu biti

4. *Todo* – komentari za budući rad koji se nalaze na mestima gde funkcionalnosti nisu u potpunosti završene, ili na mestima gde se predviđa neka nova/napredna funkcionalnost koja još uvek nije implementirana. Ovi komentari takođe ukazuju na bagove koje treba analizirati i popraviti. Na primer:

```
/**
 * TODO: Ovde treba dodati jos dve funkcije za dohvatanje i ispisivanje
 *
 */
```

5. *Notice* – komentari koji predstavljaju upozorenja i napomene namenjene programerima i korisnicima programskog koda. Mogu da se odnose na zastarele delove koda i instrukcije o alternativnim metodama ili klasama koje treba koristiti, da daju primere korišćenja/pozivanja nekih delova koda, da upozoravaju na određene izuzetke. Oni takođe objašnjavaju da je nešto implementirano na određeni način zbog бага ili znanog problema. Na primer:

```
/**
 * Ovaj deo koda bi mogao lepše da se realizuje.
 * Potrebno je dodati za svaku funkciju povratnu vrednost shodno
 * novom standardu PHP8.
 */
```

6. *General* – opšti komentari koji sadrže podatke u autorstvu, licenci, korišćenim bibliotekama, verziji koda, nazivu fajla i sl. Na primer:

```
/**
 * DataFrameHandler.cpp
 * Datum: 23.5.2022.
 * @author D.Draskovic <drazen@etf.bg.ac.rs>
 * Ovde je koriscena biblioteka CKE editor za laksi unos obavestenja.
 */
```

7. *Code* – komentari ove kategorije predstavljaju iskomentarisane i izbačene linije programskog koda što se obično radi kod debugovanja i testiranja ili kao način čuvanja starih varijanti koda. Na primer:

```
/**
 * int sum(int a, int b) {
 *     return a + b;
 * }
 */
```

8. *IDE* – komentari koji su namenjeni za komunikaciju sa integrisanim razvojnim okruženjem ili kompajlerom. Obično su automatski generisani i nisu značajni za ljude. Na primer:

```
// CHECKSTYLE:OFF
```