Score: 0.000 (=0.0%)
Id: 45043
Upišite niz znakova koji nedostaje (tj. označen je s ##########) u navedenom CSS kodu?

```
#header h1 {
  animation-duration: 2s;
  animation-name: ImeAnimacije;
  animation-iteration-count: 10;
  animation-timing-function: linear;
  animation-direction: alternate;
}

########## ImeAnimacije {
  from {
     margin-left: 100%;
  }
  to {
     margin-left: -100%;
  }
}
```

Student's answer:

```
class
```

Hint: Neispravno.

Correct answer:

@keyframes

Score: 1.000 (=100.0%)
Id: 45081
Kako ćemo, slijedeći **BEM** metodologiju, postaviti **crvenu** boju teksta naslova (title) gumba (button)?

a

```
#button-title {
   color: red;
}
...
<button>
   <span id="button-title">Button title</span>
</button>
```

b

```
.btn {
    color: red;
}
...
<button class="btn">
    <span>Button title</span>
</button>
```

c

```
.button--title {
    color: red;
}
...
<button>
   <span class="button--title">Button title</span>
</button>
```

d

```
<button>
    <span style="color:red;">Button title</span>
</button>
```

e

```
.btn .title {
    color: red;
}
...
<button class="btn">
    <span class="title">Button title</span>
</button>
```

f

```
.text-danger {
    color: red;
}
<button>
    <span class="text-danger">Button title</span>
</button>
```

Score: 1.000 (=100.0%)
Id: 45127
Koja od ponuđenih tvrdnji u vezi SERVER_PUSH mehanizma u HTTP/2 protokolu je ispravna?

a Poslužitelj unaprijed šalje tokove podataka klijentu bez HTTP zahtjeva klijenta. Poslužitelj prvo šalje PUSH_PROMISE okvir da bi dojavio namjeru slanja daljnjih tokova. Klijent uvijek prihvaća SERVER_PUSH.

b Poslužitelj šalje podatke klijentu i ako PUSH_PROMISE nije zaprimljen.

c Poslužitelj uvijek šalje tokove podataka klijentu sa iti bez HTTP zahtjeva klijenta.

d Poslužitelj unaprijed šalje tokove podataka klijentu bez HTTP zahtjeva klijenta. Poslužitelj prvo šalje PUSH_PROMISE okvir da bi dojavio namjeru slanja daljnjih tokova. PUSH_PROMISE mora biti zaprimljen da bi se izbjegli višestruki klijentski zahtjevi za istim resursima. Klijent može odbiti SERVER_PUSH slanjem RST_STREAM okvira ako su resursi već u njegovoj privremenoj memoriji.

e Poslužitelj u svojoj privremenoj memoriji zapisuje identifikatore resursa koje ima svaki klijent koji mu je ikad poslao HTTP zahtjev.

Score: 0.250 (=25,0%)
Id: 45085

Koje su od sljedećih izjava **istinite**?

a Kod običnog ("*vanilla*") CSS-a i klasičnog *semantic* CSS pristupa tipično moramo pisati sav CSS kod ispočetka ("*from scratch*") odnosno ne možemo ga ponovo iskoristiti u većoj mjeri za druge projekte

b Razvoj pomoću radnih okvira (Tailwind, Bootstrap) tipično donosi nepotrebni višak programskog koda (*overhead code*)

c Radni okviri s pomoćnim klasama (*utility frameworks*, npr. Tailwind) nam omogućuju da slijedimo dobre prakse prilikom razvoja

d Obični ("*vanilla*") CSS nam omogućuje brži razvoj nego da koristimo radne okvire

e Web aplikacije razvijene u sveobuhvatnim radnim okvirima (*component frameworks*, npr. Bootstrap) imaju nedostatak da "sve izgledaju isto"

f Obični ("*vanilla*") CSS nam pomaže da pratimo dobre prakse prilikom razvoja

Score: 1.000 (=100.0%)
Id: 44937
Pretpostavimo da dvije aplikacije (X i Y) koriste uslugu AAI@Edu.hr i protokole OAuth2/OIDC. Prilikom prijave u aplikaciju X i uspješnog unosa korisničkom imena i lozinke na usluzi AAI, prijavljeni smo u aplikaciji X. Nakon toga u istom pregledniku posjetimo aplikaciju Y te se želimo u njoj prijaviti.

Što od navedenog je istina?

a Prijavom na aplikaciju Y bit ćemo automatski odjavljeni iz aplikacije X

b Budući da i X i Y koriste AAI@Edu.hr one dijele cookie, pa je cookie aplikacije X odmah prepoznati u aplikaciji Y, čime smo automatski prijavljeni.

c Aplikacija Y će nas preusmjeriti na uslugu AAI@Edu.hr te ćemo se morati ponovo prijaviti, jer svaka od aplikacija ima vlastiti cookie.

d Budući da i X i Y koriste AAI@Edu.hr, moramo se prvo odjaviti iz aplikacije X da bi se mogli prijaviti u Y.

e Aplikacija Y će nas preusmjeriti na uslugu AAI@Edu.hr, ali ćemo odmah biti preusmjereni natrag i biti prijavljeni na aplikaciju Y kao posljedicu činjenice da smo od ranije imali valjani cookie za AAI.

Score: 1.000 (=100.0%)
Id: 44914
Zamislite sljedeću hipotetsku situaciju:

Napisati ste mobilnu aplikaciju u koju se korisnici prijavljuju koristeći AAI@EduHr, nakon čega aplikacija šalje token (kojeg je izdao AAI@EduHr) Edgaru tražeći od Edgara da vam prikaže trenutni broj bodova na predmetu.

U kontekstu korištenja vanjske usluge za autentifikaciju, što je AAI@EduHr?

a Resource server

b Authorization server

c Proxy

d Resource owner

e Client

Score: -0.250 (=-25.0%)
Id: 45059
Kod sigurnosnog nedostatka loša autentifikacija, koji pristup koriste vertikalni napadi:

a cijeli rječnik zaporki za jednog korisnika (npr. sa ili admin)

b slabe zaporke iz rječnika (kratke ili niske entropije)

c *brute force*

d jedna lozinka za sve korisnike

e napredni automatizirani alati

Score: -0.250 (=-25.0%)
Id: 45051
Koji sigurnosni propust napadač želi iskoristiti ako koristi sljedeću proceduru napada:

1. Napadač kreira hiperlink koji, osim URL-a legitimnog poslužitelja, sadrži i zlonamjerni skriptni kod.
2. Napadač šalje zlonamjerni hiperlink korisniku (npr. putem elektroničke pošte).
3. Korisnik aktivira hiperlink, pri čemu se legitimnom web poslužitelju koji sadrži sigurnosni propust šalje HTTP zahtjev za ranjivom web stranicom.
4. Legitimni web poslužitelj šalje korisniku ranjivu web stranicu kao HTTP odgovor. Zlonamjerni skriptni kod nije umetnut u poslanu web stranicu, nego je još uvijek sadržan samo unutar hiperlinka.
5. Korisnikov web preglednik interpretira ranjivu web stranicu koja se sada nalazi na lokalnom korisnikovom sustavu. Nailaskom na ranjivi dio stranice, aktivira se zlonamjerni skriptni kod iz hiperlinka (kao vrijednost jednog od parametara dobivene web stranice), koji se potom izvršava s ovlastima web preglednika unutar lokalne zone korisnikovog računala.

a Trajni (pohranjeni) XSS sigurnosni propust

b Loša autentifikacija

c Lokalni (DOM) XSS sigurnosni propust

d Jednokratni (reflektirani) XSS sigurnosni propust

e Lažiranje zahtjeva na drugom sjedištu

Score: -0.250 (=-25.0%)
Id: 45071
Koja strategija se preporuča za otklanjanje ranjivosti loša kontrola pristupa?

a inverzni Turingov test (CAPTCHA)

b filtriranje IP adresa

c sanitizacija unosa od strane korisnika

d dodati neku tajnu (token), a ne prihvaćati sve podatke automatski

e zamjena javno dostupnih internih referenci s privremenim vrijednostima koje se na poslužitelju preslikavaju u prave

Score: 0.500 (=50.0%)
Id: 45048
Označite sve točne tvrdnje vezane za *XML External Entity* (XXE) sigurnosni nedostatak

a Ranjive su sve aplikacije koje parsiraju XML datoteke

b U otklanjanju ovog nedostatka potrebno je ograničiti veličinu učitane XML datoteke

c Potrebno je izbjegavati korištenje složenijih XML struktura ako nisu potrebne

d Učitavanje vanjskih XML datoteka je štetno i potrebno ga je onemogućiti

e Potrebno je proučiti i ažurirati postavke XML parsera za učitavanje vanjskih entiteta

Score: 1.000 (=100.0%)
Id: 45057
Web aplikacija za autentifikaciju koristiti sljedeći SQL kôd:

```
String SQLQuery = "SELECT Username FROM Users WHERE Username = '"
+ username + "' AND Password = '" + password + "'";
```

Koje nizove znakova napadač mora unijeti za vrijednosti varijabli username i password kako bi se mogao prijaviti na sustav kao prvi korisnik u korisničkoj tablici `Users`?

a `' OR ''='` i `''`

b `' OR ''='` i `';'`

c `' OR ''='` i `' OR ''='`

d `'*'` i `'*'`

e `' OR ''='` i `'*'`

Score: 0.000 (=0.0%)
Id: 45046
Unesi CSS koji će nastati prevođenjem sljedećeg SASS (.scss) koda:

```
%default {
    color: black;
}
%alternative {
    color: red;
}
$brand-color: blue;

h1 {
    @extend %default;
    font-size: 2rem;
}

h2 {
    @extend %default;
    font-size: 1rem;
}
```

(nemojte upisivati komentare, indentacija i novi redovi će biti noramlizirani prije evaluacije)

Ako smatrate da je kod neispravan, upišite "error".

Student's answer:

Hint: Your answer is not correct:

Correct answer:

```
h2, h1 {
  color: black;
}

h1 {
  font-size: 2rem;
}

h2 {
  font-size: 1rem;
}
```

Score: -0.250 (=-25.0%)
Id: 44920
Zamislimo hipotetsku situaciju u kojoj bi za prijavu u Edgar umjesto preusmjeravanja na stranici sustava AAI@Edu.HR, vaše korisničko ime i lozinku za AAI upisali u formu na Edgaru i prepustili Edgaru da obavi prijavu umjesto vas.

Koji OAuth2/OIDC tok bi se koristio?

a Authorization Code Flow + PKCE

b Implicit flow

c Authorization Code Flow, ali bez PKCE

d Resource Owner Password Flow

e Client Credentials Flow

Score: -0.250 (=-25.0%)
Id: 44911
Zaglavlje kojim se šalje token poslužitelju ima oblik

a `Authentication: Token sadržaj_tokena`

b `Cookie: token=sadržaj_tokena`

c `Authorization: Token sadržaj_tokena`

d `Authentication: Bearer sadržaj_tokena`

e `Authorization: Bearer sadržaj_tokena`

Score: 1.000 (=100.0%)
Id: 45069
Koja strategija se preporuča za otklanjanje ranjivosti lažiranje zahtjeva na drugom sjedištu?

a koristiti HTTP POST umjesto HTTP GET

b filtriranje IP adresa

c sanitizacija unosa od strane korisnika

d eliminacija izravnih referenci s privremenim neizravnim vrijednostima

e inverzni Turingov test (CAPTCHA)

Score: 1.000 (=100.0%)
Id: 44907
Ako se za provjeru autentičnosti koristi mehanizam *Digest Authentication* korisnik se može odjaviti iz aplikacije

a samo zatvaranjem preglednika

b brisanjem svih kolačića u svom pregledniku

c pozivom postupka na serveru koji će u zaglavlju odgovora imati zaglavlje

```
Set-Cookie: digest=invalidate; path=/; expires=Thu, Jan 01 1970 00:00:00 UTC;
```

d pozivom postupka na serveru koji će vratiti status 403

e pozivom postupka na serveru koji će u zaglavlju odgovora imati zaglavlje

```
Set-Cookie: authentication=null; path=/;
```

Score: 0.000 (=0.0%)
Id: 44921
Označite sve vrste tokena koji se koriste u OAuth2/OIDC?

a code token

b logout token

c id token

d refresh token

e access token

Score: 1.000 (=100.0%)
Id: 45052
Ako se na poslužitelju ne koriste sigurne reference na objekte pa je osim pristupa putanji:

`/user/getAccounts`

logiranom korisniku bez administratorskih ovlasti dozvoljen pristup putanji:

`/admin/getAccounts`

čime korisnik dobiva administratorske ovlasti, o kojem sigurnosnom nedostaku se radi?

a Nesigurna pohrana osjetljivih podataka

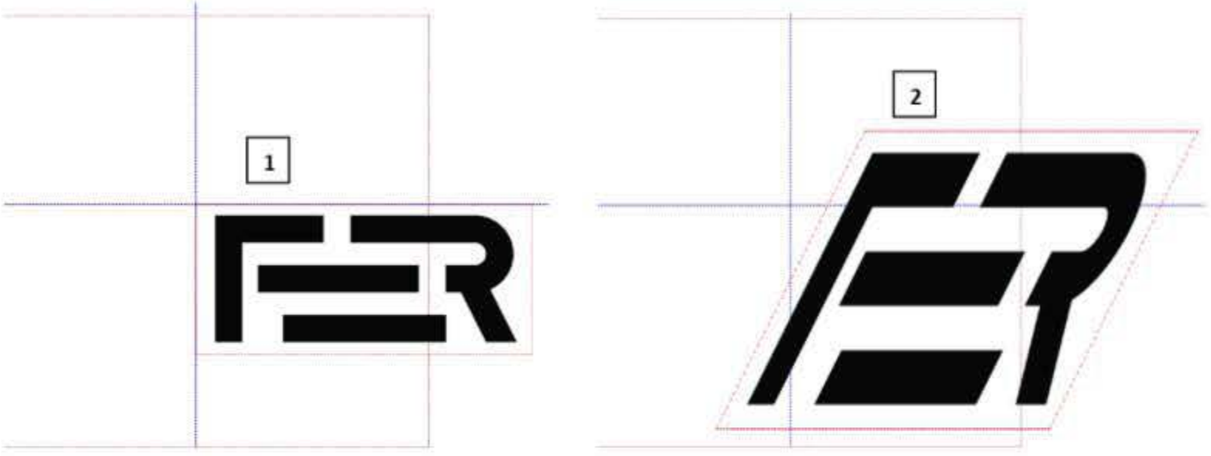
b XSS sigurnosni nedostatak

c Loša autentifikacija

d Lažiranje zahtjeva na drugom sjedištu

e Loša kontrola pristupa

Score: 0.000 (=0.0%)
Id: 45042
Kojim CSS izrazom možemo logo FER-a (sliku) iz inicijalnog stanja 1 dovesti u stanje 2?



a transform: scale(2, 1) skew(-45deg, 0deg);

b transform: scale(2, 1) skew(45deg, 45deg);

c transform: scale(1, 2) skew(0deg, 45deg);

d transform: scale(1, 2) skew(-45deg, 0deg);

e transform: scale(2, 1) skew(-0deg, -45deg);

f transform: scale(1, 2) skew(45deg, 45deg);

g niti jedan od navedenih izraza

Score: -0.250 (=-25.0%)
Id: 44922
Podaci unutar identifikacijskog token u OpenId Connect protokolu nazivaju se

a permissions

b claims

c restrictions

d scopes

e identities