

برنامه‌سازی وب

آزمون پایاترم - نیم‌سال پاییز ۱۴۰۴

نام استاد: علی ابریشمی

دانشگاه صنعتی شریف

زمان آزمون: ۳۰+۲۱ دقیقه

پیش از پاسخگویی به سوالات حتماً به موارد زیر توجه کنید.

- مطابق دستورنامه آموزش بابت امتحانات مجازی، می‌بایست طی مدت امتحان در کلاس مجازی حضور یابید. سوالات خود را پس از گذشت ساعت اول در همان کلاس مجازی بپرسید. به سوالات مطرح شده در سایر پلتفرم‌ها پاسخی داده نخواهد شد.
- پاسخ‌هایتان را برای سوالات بخش مفهومی و طراحی به صورت دستنویس آپلود کنید. برای بخش عملی نیز پاسخ‌هایتان باید به صورت تایپ‌شده باشند. در نهایت ۳ فایل PDF برای هرکدام از بخش‌های سه‌گانه‌ی آزمون از شما تحویل گرفته خواهد شد. تاکید می‌شود بخش عملی را باید به صورت تایپ‌شده تحویل دهید. چرا که این بخش توسط مصححان به صورت دآوری خودکار و با تست‌های خودکار تصحیح خواهد شد (طبیعتاً نه در زمان خود آزمون) و به پاسخ‌های دست‌نویس این بخش نمره‌ای تعلق نخواهد گرفت. فرمت فایل تحویلی برای هر بخش نیز باید به صورت زیر باشد:
WP-FINAL-SEC[SECID] - [STDID].pdf
که در آن SECID شماره بخش (۱ یا ۲ یا ۳) و STDID شماره دانشجویی شماست. نهایتاً این سه فایل را به صورت یک فایل زیپ با نام زیر بارگذاری کنید:
WP-FINAL-[STDID].zip
- بدیهیست که پاسخ‌های دستنویس باید خوانا و واضح باشند و مسئولیت هر کسر نمره به دلیل ناخوانا یا ناواضح بودن (تار بودن عکس، دست‌خط بد، ...) متوجه شخص دانشجوست.
- پیش از امتحان از فراهم بودن لوازم و شرایط آزمون (شارژ کافی دستگاه‌ها، اینترنت پشتیبان، و ...) اطمینان حاصل فرمایید. با این حال در صورت وقوع هرگونه اختلال در حین آزمون، مطابق اعلامیه و شیوه‌نامه جدید آموزش عمل کنید.
- در درس‌های نرم‌افزاری، دقیق و کامل پاسخ دادن به سوالات نقش کلیدی در گرفتن نمره پاسخ صحیح دارد. لذا از درج پاسخ‌های ناقص، مبهم و نامفهوم اکیداً پرهیزید چرا که به این پاسخ‌ها نمره‌ای تعلق نخواهد گرفت.
- استفاده از هوش مصنوعی برای تحقیق و جستجو مجاز است، اما محول کردن نوشتن پاسخ سوالات (متن، نمودار و ...) اکیداً ممنوع بوده و در صورت تشخیص نمره‌ی سوال برای شما صفر در نظر گرفته می‌شود.
- ارجاع به منابع تکمیلی و کمکی قرار داده شده در کونرا نه تنها ممنوع نبوده بلکه تشویق هم می‌شود.
- مطابق آیین‌نامه‌ی آموزشی، همکاری و هم‌فکری به امتحانات پایان‌ترم اکیداً ممنوع بوده و در صورت کشف هرگونه تقلب، با هر دو طرف خاطی مطابق شیوه‌نامه‌های آموزش برخورد خواهد شد.
- مدت زمان آزمون در اصل ۳ ساعت و نیم است. با این حال نیم ساعت بیشتر در نظر گرفته شده تا شما بتوانید با آرامش پاسخ‌های خود را بارگذاری نمایید. توجه فرمایید این زمان به هیچ‌وجه تمدید نخواهد شد؛ لذا از درخواست تمدید اکیداً پرهیزید. توصیه می‌شود مدیریت زمان خود را مطابق زمان‌های پیشنهادی انجام دهید.
- پس از پایان زمان آزمون، هر نیم ساعت تاخیر در بارگذاری پاسخ‌ها ۱۰ درصد از نمره کل آزمون شما کم خواهد کرد. تا حداکثر ۲ ساعت پس از پایان زمان آزمون می‌توانید با تاخیر بارگذاری کنید و پس از آن مهلت آزمون کاملاً تمام خواهد شد.

موفق باشید.

۱ سوالات مفهومی (۳۵ نمره) – زمان پیشنهادی: ۴۵ دقیقه

۱. در بین موارد زیر آتھایی که الگوها و استانداردهای گفته شده طی درس را رعایت میکنند (الگوها یا Patterns) را با P و بقیه را (پادالگوها یا Anti-patterns) با A نشان دهید. همچنین دلیل مختصر خود را برای هر مورد بنویسید. (۸ نمره)

گزاره	A/P	دلیل
شخصی اصرار دارد در یک بک‌اند نوشته شده با DRF از هیچ چیز جز کلاس <code>APIView</code> استفاده نکند.		
یک توسعه‌دهنده‌ی React در قسمتی از یک صفحه، مستقیماً استیت‌ها را بین کامپوننت‌ها پاس می‌دهد		
یک بک‌اند URL به صورت <code>/tasks/update-existing-task/{id}</code> دارد.		
کاربرانی که وارد سامانه شده‌اند، در صورت درخواست منبعی که به آن دسترسی ندارند کد ۴۰۳ دریافت می‌کنند		

۲. مفهوم اصطلاحات زیر را مختصراً (ولی با حفظ دقت) توضیح دهید. (۵ نمره)

۱. RBAC

۲. Component-based Design

۳. Layered Architecture

۴. Server-side Rendering (SSR)

۵. State Hoisting

۳. تعدادی خط کد یا کامند در جدول زیر آمده است. توضیح دهید که محیطی که این خط در آن وجود دارد و در حال اجراست محیط توسعه (dev) است یا محیط استقرار اصلی (prod). (فرض کنید تمام استانداردها در کدبیس مربوطه رعایت شده‌اند). همچنین علت انتخاب خود را توضیح دهید. (۸ نمره)

گزاره	prod/dev	علت
<code>CORS_ALLOW_ALL_ORIGINS = True</code>		
<code>ALLOWED_HOSTS = [A, B, ...]</code>		
<code>python manage.py runserver</code>		
<code>SECRET_KEY = "django-insecure-..."</code>		

۴. در DRF، کارکرد هرکدام از متدهای `has_permission` و `has_object_permission` را با ذکر یک مثال از یک پروژه یا محصول دلخواه شرح داده و تفاوت‌هایشان را توضیح دهید. (۷ نمره)

۵. توضیح دهید چگونه در یک Back-end نوشته شده با DRF می‌توان یک Endpoint آپلود فایل صوتی در یک پروژه دلخواه (برای مثال یک فایل با فرمت mp3) داشت. موارد زیر را در پاسخ خود بیاورید: (۷ نمره)

- Headerهای مهم درخواست
 - محل ذخیره فایل صوتی
 - یک URL نمونه برای Endpoint مربوطه
 - آنچه در پایگاه داده ذخیره می‌شود.
 - فیلد مدل لازمه در کلاسی که در `models.py` می‌آید.
- پاسختان باید مختصر اما دقیق باشد.

۲ سوالات عملی (۳۵ نمره) – زمان پیشنهادی: ۷۵ دقیقه

۶. (۱۲ نمره)

با توجه به دو مدل زیر در جنگو:

- Customer

- Order

یک کوئری تک خطی (one-liner) با استفاده از Django ORM بنویسید که نام تمام مشتریانی را برگرداند که همه شرایط زیر را داشته باشند:

۱. آخرین سفارش آنها (بر اساس فیلد created_at) دارای وضعیت REFUNDED باشد.

۲. در ۹۰ روز گذشته حداقل ۳ سفارش با وضعیت COMPLETED داشته باشند.

۳. هیچگاه سفارشی با وضعیت CHARGEBACK نداشته باشند.

نکات و محدودیت‌ها:

- کوئری باید دقیقاً در یک خط نوشته شود.

- استفاده از Subquery، OuterRef، Exists و annotate مجاز است.

کد مدل‌ها:

```
from django.db import models

class Customer(models.Model):
    email = models.EmailField(unique=True)

class Order(models.Model):
    class Status(models.TextChoices):
        PENDING = "PENDING"
        COMPLETED = "COMPLETED"
        REFUNDED = "REFUNDED"
        CHARGEBACK = "CHARGEBACK"

    customer = models.ForeignKey(Customer, related_name="orders",
                                on_delete=models.CASCADE)
    status = models.CharField(max_length=20, choices=Status.choices)
    total_cents = models.PositiveIntegerField()
    created_at = models.DateTimeField(db_index=True)
```

در نظر بگیرید یک برنامه React شامل دو کامپوننت است که یکی لیستی از کاربران را نمایش می‌دهد و دیگری کامپوننت اصلی برنامه است. این برنامه از `useState` و `useEffect` برای مدیریت وضعیت و دریافت داده‌ها استفاده می‌کند.

بدون تغییر ساختار TSX File و بدون اضافه کردن state جدید، به سؤالات زیر پاسخ دهید:

۱. توضیح دهید چرا با کلیک روی یک کاربر، رابط کاربری ممکن است رفتار غیرقابل پیش‌بینی داشته باشد یا در برخی مواقع اصلاً `re-render` نشود.

۲. با حداقل تغییر ممکن در خطوط کد، پیاده‌سازی را به‌گونه‌ای اصلاح کنید که:

- رفتار برنامه کاملاً قطعی (deterministic) شود،
- از `re-render`‌های غیرضروری جلوگیری شود،

۳. توضیح دهید چرا راه‌حل شما از نظر مفاهیم زیر درست عمل می‌کند:

- تغییرناپذیری داده‌ها (immutability)
- React در closures
- فاز `render` و قوانین آن

نکته: انتظار می‌رود پاسخ شما مبتنی بر درک عمیق از مدل اجرایی React باشد، نه صرفاً اصلاح ظاهری کد.

قسمت‌های توضیحی این سوال را می‌توانید به صورت تایپ‌شده هم ارسال کنید.

```
import * as React from "react"

type User = {
  id: string
  name: string
  score: number
}

type UserListProps = {
  users: User[]
  onSelect: (user: User) => void
}

function UserList({ users, onSelect }: UserListProps) {
  const sorted = users.sort((a, b) => a.score - b.score)

  return (
    <ul>
      {sorted.map((u) => (
        <li key={u.id} onClick={() => onSelect(u)}>
          {u.name}
        </li>
      ))}
    </ul>
  )
}
```

```

export default function App() {
  const [selected, setSelected] = React.useState<User | null>(null)
  const [users, setUsers] = React.useState<User[]>([])

  React.useEffect(() => {
    fetch("/api/users")
      .then((r) => r.json() as Promise<User[]>)
      .then(setUsers)
  }, [])

  const onSelect = (user: User) => {
    setSelected(user)
    setUsers(users)
  }

  return (
    <>
      <UserList users={users} onSelect={onSelect} />
      <div>{selected?.name}</div>
    </>
  )
}

```

۸. (۵ نمره - امتیازی)

فرض کنید در یک برنامه نوشته شده با تایپاسکریپت نوع‌های زیر در اختیار شما قرار دارد:

- Command
- CommandType
- CommandResult

هدف شما طراحی فقط type / interface های عمومی (بدون پیاده‌سازی) برای یک API TypeScript نوع-امن (type-safe) است که «فرمان‌ها» را به handler های مناسب وصل می‌کند و نتیجه درست را برمی‌گرداند.

این API باید شامل متدهای زیر باشد:

- handle(type, handler)
- unhandle(type, handler)
- dispatch(command)

شرایط:

۱. اگر type یک مقدار مشخص باشد (مثلاً "CreateUser")، پارامتر handler باید دقیقاً همان نوع Command متناظر را دریافت کند و دقیقاً همان CommandResult متناظر را برگرداند.

۲. اگر type برابر "*" باشد، handler باید یک Command عمومی دریافت کند و یک CommandResult عمومی برگرداند.

۳. متد dispatch باید فقط اجازه ارسال Command را بدهد و نوع خروجی آن باید بر اساس command.type به‌درستی محدود شود.

۴. استنتاج نوع (type inference) باید درست عمل کند:

- handle("CreateUser", c => c.email) باید type-check شود.
- handle("CreateUser", c => c.userId) نباید type-check شود.
- اگر t: CommandType باشد، آنگاه (t, c => ...) باید باعث شود c از نوع Command باشد و خروجی handler از نوع CommandResult.
- اگر const t = "CreateUser" as const باشد، نوع ورودی و خروجی handler باید به‌درستی محدود شود.
- dispatch(type: "DeleteUser", userId: "x") باید خروجی‌ای از نوع نتیجه متناظر "DeleteUser" داشته باشد.

۵. استفاده از Extract, any و assertion های عمومی مانند as Foo مجاز نیست.

```
type Command =
| { type: "CreateUser"; email: string }
| { type: "DeleteUser"; userId: string }
| { type: "ListUsers"; limit?: number }

type CommandType = Command["type"]

type CommandResult =
| { type: "CreateUser"; userId: string }
| { type: "DeleteUser"; ok: true }
| { type: "ListUsers"; users: Array<{ userId: string; email: string }> }
```

خروجی مورد انتظار: تعریف type / interface لازم برای تابعی شبیه به createRouter() که یک شیء با API فوق برمی‌گرداند، به‌طوری که تمام محدودیت‌ها فقط توسط سیستم type در TypeScript تضمین شوند.

۳ سوالات طراحی (۳۵ نمره) – زمان پیشنهادی: ۹۰ دقیقه

راهنمایی پاسخگویی به سوالات طراحی

- در مهندسی نرم افزار، تفاوت معماری و طراحی در سطح جزئیات است که طراحی به جزئیات بیشتری می پردازد. لذا انتظار می رود جزئیات لازمه را با دقت هرچه تمام تر در نظر بگیرید. این در نظر گرفتن باید در پاسخ های شما مشهود باشد.
- پاسخ های شما باید کامل و دقیق نوشته شوند. فرض کنید این طراحی قرار است به دست یک توسعه دهنده ی نرم افزار برسد. او باید بتواند با خواندن پاسخ شما، دقیقاً آنچه طراحی کردید را بدون ذره ای تغییر پیاده کند.
- ملاک نمره دهی، غیر از رعایت و در نظر گرفتن خواسته ها و محدودیت های سوالات، فاقد ابهام بودن پاسخ شماست. فرضاً که خواسته ها و محدودیت ها را در نظر گرفته باشید، در صورتی نمره کامل خواهید گرفت که سوال یا ابهامی برای توسعه دهنده ای که قرار است آن را پیاده سازی کند وجود نداشته باشد.
- پیشنهاد می شود سامانه ها و برنامه های خواسته شده را به تعدادی زیرمساله بشکانید، سپس در هر بخش از پاسخ خود سعی کنید بر روی همان زیرمساله تمرکز کنید و از آن منحرف نشوید.
- از خلاقیت های درست و به جا استقبال می شود و در صورت مشاهده در کسب نمره کامل شما را یاری خواهد کرد.

۹. (۱۵ نمره)

فرض کنید می خواهید یک سیستم مسیریابی شبیه React Router DOM را از ابتدا و بدون استفاده از هیچ کتابخانه آماده ای طراحی کنید تا بتوان مسیرهای یک برنامه تک صفحه ای (SPA) React را مدیریت کرد.

در پاسخ خود، به صورت مفهومی و طراحی محور به موارد زیر بپردازید:

- به طور کلی توضیح دهید سیستم مسیریابی شما چگونه مسیر فعلی برنامه را تشخیص می دهد و بر اساس آن تصمیم می گیرد کدام کامپوننت باید رندر شود.
- اجزای اصلی یا API های مهم این سیستم (مانند کامپوننت روتر، تعریف مسیرها، لینک ها یا هوک های ناوبری) را معرفی کنید و نقش هر کدام را به اختصار توضیح دهید.
- توضیح دهید تغییر مسیر (ناوبری) در اثر کلیک کاربر یا دستور برنامه چگونه انجام می شود، به گونه ای که آدرس URL تغییر کند اما صفحه دوباره بارگذاری نشود.
- نقش هوک ها را در پاسخ خود مشخص کنید. آن ها چگونه به شما در طراحی کمک کرده اند؟

نکات:

- از ذکر یا استفاده از کتابخانه های آماده مسیریابی خودداری نمایید.
- نیازی به نوشتن کد نیست؛ توضیح مفهومی یا شبه کد کافی است.
- پاسخ باید شفاف، منسجم و قابل فهم باشد.
- تایپ ها، کامپوننت ها، states و props باید در جوابتان دقیقاً مشهود باشند.

سامانه CW دانشگاه صنعتی شریف که بر پایه ابزار Moodle پیاده‌سازی شده است، به دلیل عدم به‌روزرسانی منظم و همچنین اعمال تغییرات مستقیم روی کد منبع (به جای استفاده از Package ها و افزونه‌های رسمی)، در فرآیند به‌روزرسانی با تداخل‌های جدی (Conflict) مواجه شده است؛ به گونه‌ای که ادامه نگهداری و توسعه آن از نظر مهندسی مقرون به صرفه نیست.

در نتیجه، تیم فنی ICTC تصمیم گرفته است این سامانه را از ابتدا و با استفاده از فریم‌ورک Django و با استفاده از Django REST Framework بازطراحی و پیاده‌سازی کند، چرا که نگهداری، توسعه و مقیاس‌پذیری آن را ساده‌تر از ادامه کار با کد قدیمی PHP می‌دانند. فرض کنید شما عضو تیم طراحی معماری این پروژه هستید. محصول جدید باید فرآیندهایی که CW فعلی دارد را (از جمله تعریف و تحویل تکلیف، آزمون، بارگذاری فایل، تشکیل و مدیریت کلاس، اعلان‌ها و فضای پرسش و پاسخ برای تکالیف) داشته باشد.

- فهرستی از Django App های اصلی مورد نیاز برای این سامانه ارائه دهید و شرح مختصری از مسئولیت و کارکرد هر App بیان کنید.

- برای هر App تمامی Entity ها را معرفی نمایید.

- برای هر Entity:

۱. فیلدهای اصلی آن را نام ببرید،

۲. نوع داده هر فیلد را مشخص کنید،

۳. و علت وجود هر فیلد و نقش آن در سامانه را توضیح دهید.

- روابط بین Entity ها (مانند One-to-Many یا Many-to-Many) را در سطح مفهومی توضیح داده و بیان کنید چرا این روابط برای پاسخ‌گویی به نیازمندی‌های سامانه ضروری هستند.

- برای حداقل سه تا اپ‌ها view ها را بنویسید و توضیح دهید هر کدام چه کار می‌کنند. همچنین URL متصل به آن‌ها را هم بنویسید و قید کنید HTTP Method هر درخواست کدام است و برای پاسخ خود دلیل بیاورید.

- مختصراً توضیح دهید view های قید شده چگونه تست‌پذیر خواهند بود.

موفق باشید.