سوال اول:

گیت یه سیستم کنترل ورژنه که باهاش میتونیم تغییرات داده شده توی فایل‌های مختلف رو پیگیری کنیم

کلا برای انجام پروژه ها بصورت گروهی از گیت استفاده میشه و اینطوره که اگه یه فایلی از پروژه‌ای که تیم داره روش کار میکنه ویرایش بشه، گیت میتونه این رو مشخص کنه که چه کسی این فایل رو ویرایش کرده، چه چیزی از این فایل رو ویرایش کرده و ...

برای اینکه بخوایم توی دایرکتوری از گیت استفاده کنیم، توی همون دایرکتوری از ترمینال دستور git init رو میزنیم. درواقع با این دستور به دایرکتوری میگیم که تحت نظر گیت هستی

برای اینکه بخوایم وضعیت تغییر فایل هامون رو ببینیم از دستور git status استفاده میکنیم

بعد از اینکه فایل هامون رو تغییر دادیم باید اونارو ببریم توی stage. در واقع stage یه سکوی پرتاب هست. این کار رو با دستور git add filename انجام میدیم. اگه هم بخوایم کل فایل های ویرایش شده رو ببریم توی stage از دستور git add . یا git add –A استفاده میکنیم

بعد از این مرحله باید این stage رو با یه پیامی کامیت کنیم تا فرایند رفتن روی stage کامل بشه که این کار رو با دستور git commit –m ‘message’ انجام میدیم

برای اینکه بخوایم چیزی رو stage خارج کنیم، از دستور git reset filename استفاده میکنیم

برای درست کردن یک branch جدید از دستور git branch branch\_name استفاده میکنیم و برای اینکه از برنچ master بریم روی اون branch از دستور git checkout branch\_name استفاده میکنیم

برای اینکه پروژه رو روی سیستم خودمون داشته باشیم از دستور git clone address استفاده میکنیم

برای پول کردن از دستور git pull origin master استفاده میشه.(اگه بخوایم روی برنچ master پول کنیم)

برای پوش کردن هم از دستور git push branchName master استفاده میشه(اگه بخوایم روی master پوش کنیم)

سوال سوم: تفاوت بین local, sesstion storage و cookie:

کلا از این سه تا برای خواندن مجدد اطلاعات توسط browser کاربر استفاده میشه.

Session storage اینطوره که مواردی که ما میخوایم رو توی مرورگر کاربر ذخیره میکنه و حتی بعد از refresh کردن صفحه هم اونارو نگه میداره اما وقتی کلا مرورگر بسته بشه تمام اطلاعاتی که توش هستند از بین میرند و پاک میشن. برای ذخیره اطلاعاتی از کاربر که در آینده بهشون نیازی نداریم استفاده میشه. مثلا سن و ...

Local storage هم انگار اینجوریه که ورژن پیشرفته session storage هست با این تفاوت که وقتی مرورگر بسته بشه هم داده ها از بین نمیرن و بعد از مدت‌ها (یه هفته یک سال یا ....) باز هم قابل استفاده هستند. از کاربرد local storage هم توی نگه داشتن سبد خرید توی سایت های فروشگاهی و ... میشه گفت

Cookie ها بیشتر برای شناسایی کاربر استفاده میشن که شامل یه کلید و مقدار هستند که مواردی مثل token رو که مربوط به کاربر هست، روی کامپیوتر کاربر نگه میدارن و اینطوری ارتباط بین کاربر و سرور راحت تر میشه. توی همه سایت ها میشه برای شناسایی کاربر و login از cookie استفاده کرد

کلا cookie ها حجم بیشتری از اطلاعات رو میتونن در خودشون نگه دارن(نسبت به local storage) و اینکه کوکی بیشتر برای اشتراک گذاری اطلاعات بین سرور و کامپیوتر کاربر استفاده میشه اما local storage برای کم کردن تعداد درخواست‌های بین سرور و کامپیوتر اون سرور استفاده میشه

سوال چهارم:

SSR یعنی پردازش اطلاعات در سمت سرور اتفاق میفته(server side rendering)

CSR یعنی پردازش اطلاعات در سمت کلاینت و روی مرورگر کاربر اتفاق میفته(client side rendering). مثلا ابزار create-react-app از این دسته ابزارهاست که اینکار رو برای ما انجام میده

مقایسه اول درباره سرعت این دو روشه. SSR مربوط به اولین بارگذاری هست و این به این معناعه که SSR تاثیر چندانی روی سرعت بارگذاری نداره. با این اوصاف CSR عملکرد بهتری داره

مقایسه دوم درباره گرفتن اطلاعات هست. تویCSR تا جایی که به لود کردن یک ماژول نیاز نباشه، برنامه های وب میتونند بدون اینترنت اجرا بشن(درصورتیکه نیاز به یه API برای دریافت اطلاعات نباشه). توی این حالت بعد از لود، برنامه مثل یکی از برنامه های دسکتاپ کار میکنه. اما SSR اینطوری نیست و توی این روش همیشه داره درخواستی به سمت سرور فرستاده میشه که زمان بر تر از روش CSR هست

مقایسه بعدی هم توی SEO هست که به طور خلاصه، SSR گزینه بهتری هست

توی React، SSR توسط فریم‌ورک NextJs پیاده‌سازی میشه و CSR توسط NuxtJs پیاده‌سازی میشه

سوال آخر:

کارهای بزرگی که انجام دادم: الف) قبولی توی دانشگاه صنعتی شریف با کمترین امکانات

ب) بازیکن تیم آلومینیوم هرمزگان توی لیگ برتر تکواندو نوجوانان

پ) یاد گرفتن برنامه‌نویسی به تنهایی و بدون داشتن هیچ معلم یا شرکت در کلاسی