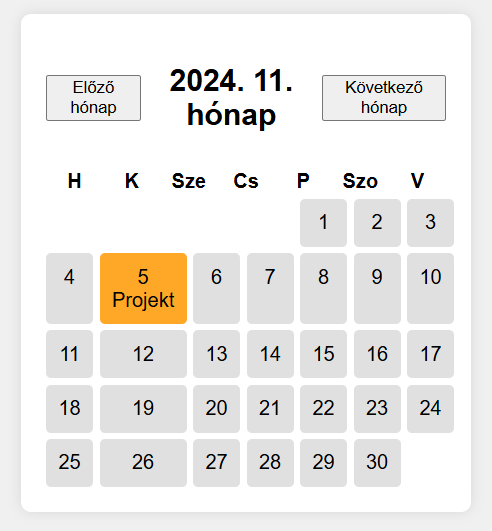
**Eseménynaptár projekt**

**A Git munkafolyamat:**

1. A mappa és a git ***Publikus repository*** neve: **esemenynaptar** legyen
2. A mappában hozzuk létre az **index.html**, valamint a **css** és a **js** fájlokat.
3. A ***main*** branch-en végezzünk el egy **Commit-ot „Kezdés” üzenettel**
4. Hozzuk létre a **developer** ágat és lépjük át erre.
5. **A fejlesztést a developer ágon végezzük!**
6. Ha készen vagyunk egy fejlesztési ütemmel, végezzük ***Commit***-ot a rá jellemző üzenettel (pl. „**1. ütem kész**”)
7. A fejlesztés elkészültével kössünk vissza a main ágra („**Merge**”)
8. Feltöltés (**Push**) a GITHUB-ra. (fejlesztés közben többször is lehet)
9. **Publikáljuk** az index.html-t a ***GITHUB***-ról

**Megvalósítandó funkciók**

1. **Naptár megjelenítése**: A JavaScript fájl dinamikusan generálja a hónap napjait és a dátumokat.
2. **Események hozzáadása és tárolása**: Az eseményeket a localStorage tárolja, hogy a felhasználók újra láthassák azokat az oldal frissítése után is.
3. **Navigáció**: A felhasználók előre és vissza lapozhatnak a hónapok között.



**1. Ütem: Projekt alapjainak létrehozása**

* **Fájlstruktúra létrehozása**: Hozz létre egy mappát a projekthez, és benne index.html, style.css, script.js fájlokat.
* **Alap HTML felépítés**: Hozd létre az index.html alapstruktúráját, például <div> elemeket a naptár fejléce és rácsának tárolására.
* **CSS fájl csatolása**: Hivatkozz a style.css fájlra az alapstílusokhoz.
* **JavaScript fájl csatolása**: Hivatkozz a script.js fájlra, hogy az eseménykezeléshez szükséges logikát könnyebben tudjuk szervezni.

**2. Ütem: HTML és CSS – A naptár kinézetének kialakítása**

* **Naptár fejléce**: Hozz létre egy fejlécet a hónap és év cím megjelenítésére, valamint navigációs gombokat ("Előző hónap" és "Következő hónap").
* **Hét napjai**: Adj hozzá egy sort, amely a hét napjait (hétfőtől vasárnapig) megjeleníti.
* **CSS rácsszerkezet beállítása**: Hozd létre a naptár rácsszerkezetét (grid elrendezés), ahol minden cella egy-egy napot képvisel.
* **Alap stílusok megadása**: Stílusozd az elemeket úgy, hogy könnyen átlátható legyen a naptár (például a napok cellái négyzet alakúak legyenek).

**3. Ütem: JavaScript – A naptár alapvető logikájának megvalósítása**

* **Dátumok kezelése**: Készíts egy funkciót, amely megjeleníti az aktuális hónap napjait a naptárban.
* **Naptár rács frissítése**: Generáld le a napok celláit dinamikusan, figyelembe véve az aktuális hónapot és évet.
* **Navigációs logika**: Implementáld a hónapváltó gombokat, hogy a felhasználók előző és következő hónapokra tudjanak léptetni.
* **Aktuális nap megjelölése**: Jelöld meg CSS-ben az aktuális napot, hogy vizuálisan is különbséget tegyünk.

**4. Ütem: Események hozzáadásának és tárolásának megvalósítása**

* **Modal létrehozása**: Készíts egy modált (felugró ablakot), ahol a felhasználó beírhatja az esemény adatait (név, dátum).
* **Form kezelése JavaScript-tel**: Készíts egy űrlapot a modálon belül, amelyben a felhasználók megadhatják az esemény nevét és dátumát.
* **Adatok mentése localStorage-ba**: Tárold az eseményeket a localStorage-ban, hogy az adatok a böngésző frissítése után is megmaradjanak.
* **Modal megjelenítése és bezárása**: Készíts eseménykezelőt, hogy a modál megjelenjen és bezáródjon, amikor a felhasználó a napokra kattint.

**5. Ütem: Események megjelenítése a naptárban**

* **Események megjelenítése**: Hozz létre egy függvényt, amely átnézi a localStorage-ban tárolt eseményeket, és megjeleníti azokat a naptár megfelelő napjain.
* **Események stílusozása**: Adj stílust az eseményeknek (pl. színes jelölő), hogy könnyebben láthatók legyenek a naptárban.
* **Események frissítése hónapváltáskor**: Gondoskodj arról, hogy a naptárban az új hónap napjainak megjelenítésekor a megfelelő események is megjelenjenek.

**6. Ütem: Tesztelés és hibajavítás**

* **Navigáció tesztelése**: Próbáld végig az előző és következő hónapra lépést, figyelve, hogy a hónapok és év változása megfelelően működjön.
* **Események hozzáadásának ellenőrzése**: Teszteld az esemény hozzáadását, hogy a helyes dátumhoz társítva jelenjenek meg az események.
* **LocalStorage tesztelése**: Ellenőrizd, hogy a localStorage megfelelően tárolja-e az eseményeket, és azok megmaradnak-e a böngésző frissítése után.
* **Reszponzivitás tesztelése**: Nézd meg, hogy kisebb kijelzőkön is jól jelenik-e meg a naptár, és szükség esetén módosítsd a CSS-t.

**7. Ütem: Fejlesztés kiegészítő funkciókkal (opcionális)**

* **Több esemény támogatása egy naphoz**: Tedd lehetővé, hogy egy naphoz több eseményt is lehessen társítani.
* **Események szerkesztése és törlése**: Adj lehetőséget az események módosítására vagy törlésére.
* **Éves naptár nézet**: Készíts egy teljes éves áttekintést, ahol egy oldalon láthatók az év hónapjai és eseményei.
* **Szűrés és keresés**: Tegyél lehetővé esemény keresését dátum vagy név alapján.