

(20%) Web 程式設計 2N 期末上機考-1

2024-6-5 at E201

Note:

1. 請不要發揮同學愛，作弊雙方除了本次考試 0 分外，平常分數另扣 20 分，情節嚴重者會送校。
2. iClass 上請繳交 md_xx.zip (md 檔，所有圖片檔，pdf 檔)，及 code_xx.zip 兩個檔案 (node_modules 請刪除)，
3. 請直接將答案寫在 f1_xx.md 上，老師實作的圖片放在 f1_htc.pdf 上，請依照老師所給的圖片來實作並標註。
4. 跟期中考相關的檔案及目錄名稱有 xx 時，必須要改成學號後 2 碼，沒有修改時，會視違犯情況扣分。
5. 每一張圖片要有機房左側背景，圖片上要有你的學號(或後兩碼)，圖片標註要跟老師所標註的類似。違者會依情節扣分。
6. 有配分的每一題，都需要 git log 至 Github，沒有傳到 Github 者，會被扣該題 50% 分數。
7. 請自評分數，將每一題的 ? 填入分數，沒有填者，不會批改，以 0 分計算。

Your (Name, ID): (207410290, 林泓君)

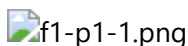
final1 題目 (100%)

- P1 (20%): 20 分
- P2 (25%): 25 分
- P3 (25%): 30 分
- P4 (30%): 0 分

總分: 75 分

注意：請依老師這次上機考所給的 code 開始撰寫，不可以寫在 w15 老師之前上課的 demo 中，因為整個檔案結構不一樣

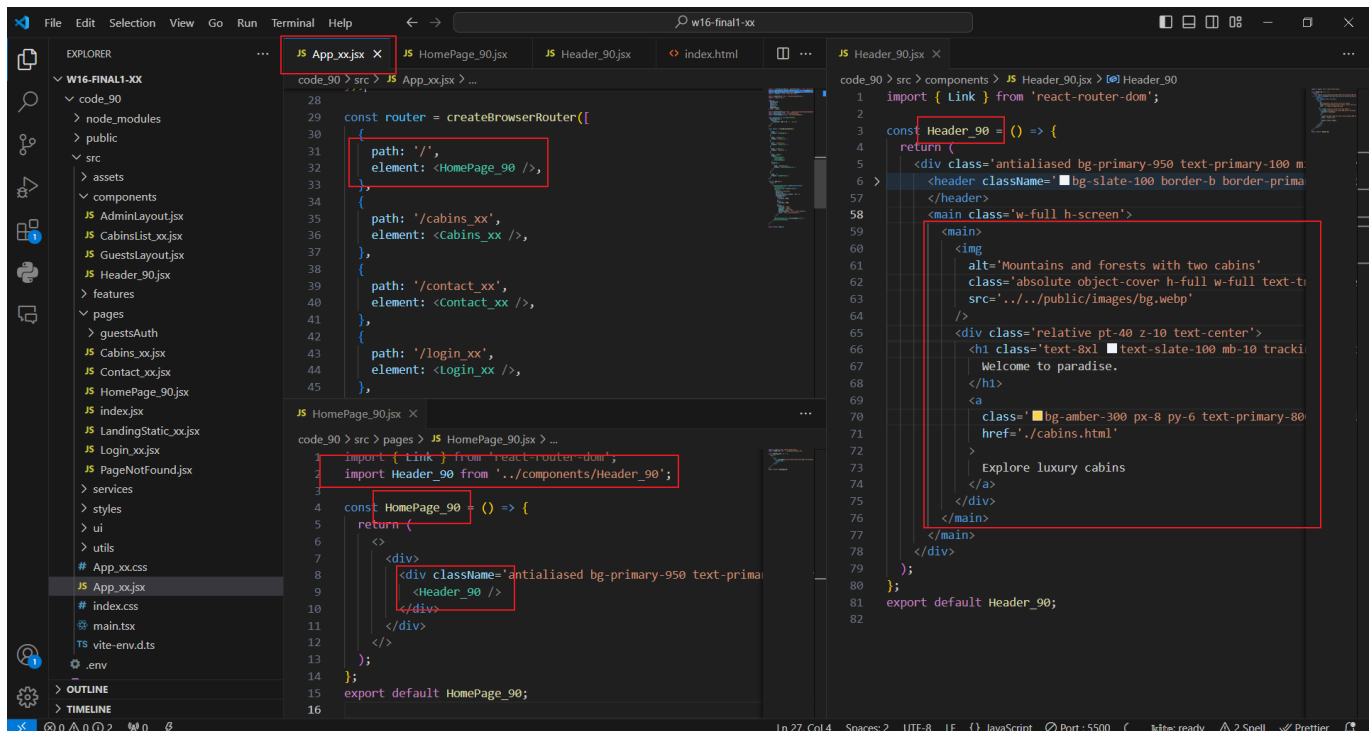
(20%) P1: 用老師所提供的 Wild Oasis theme，在瀏覽器根目錄，顯示 `<HomePage_xx />` 及本題檔案結構，如下圖

f1-p1-1.png

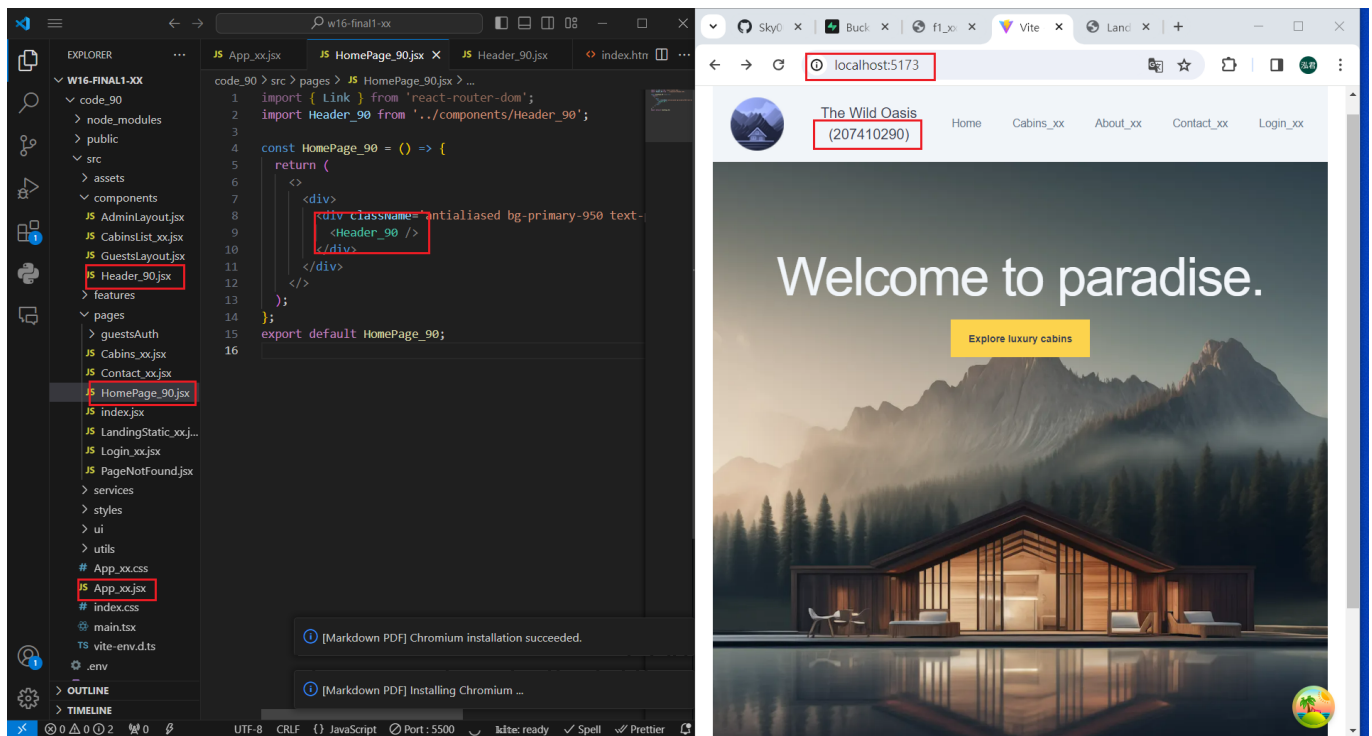
=>

Your Answer

=> `<HomePage_xx />` 及 `<Header_xx />` 顯示與圖片有關主要的 code (有些展開，有些收起)。`<Header_xx />` 在 P2, P3, P4 會用到，請寫成一個 component。



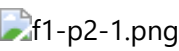
=> 顯示 App_xx.jsx 本題之路由，並透過 Chrome 路由 / 顯示圖片 (左邊 code, 右邊圖)



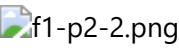
=> git log

```
git log --pretty=format:"%h%x09%an%x09%ad%x09%" --after="2024-06-04"
0c9ffe3 sky001 Wed Jun 5 21:37:51 2024 +0800 upload final answer p1,p2,p3
```

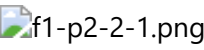
(25%) P2: 點選 P1 圖片上的 **Explore luxury cabins**，透過路由 `/cabins_xx`，可以自 **Supabase** 抓取預設 8 筆 **cabin** 資料，這與點選選單項目 **Cabins_xx** 一樣，會得到一樣的結果。顯示如下圖。



=> 在 `<Cabins_xx />` 上直接去呼叫 `useCabins()`，透過 **React Query** 去呼叫 `apiCabins.jsx` 取得 8 筆 **cabins** 資料

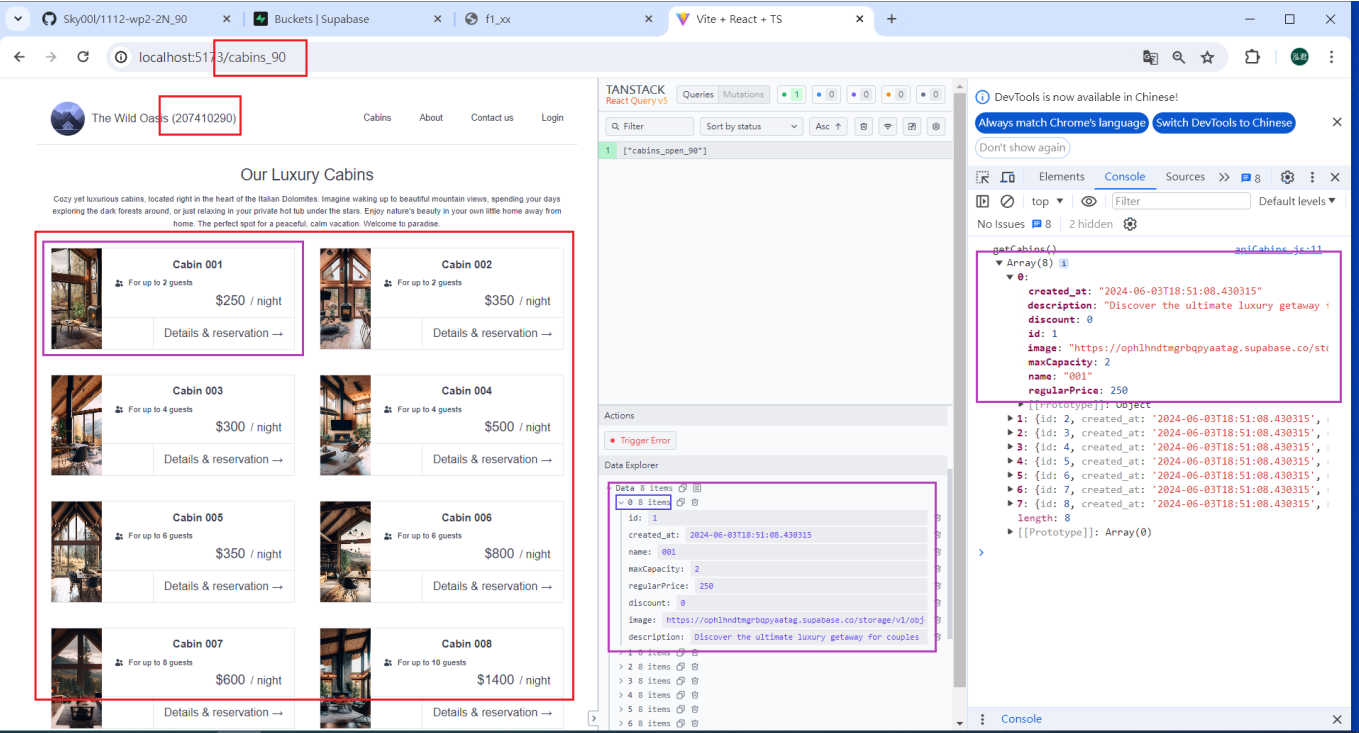


=> 目錄及檔案結構如下



Your Answer

=> **Chrome** 顯示如本題第一張圖



=> 顯示 `<Cabins_xx />` 完成本題主要的 **code**

```
code_90 > src > pages > JS Cabins_90.jsx > ...
1 import { Link } from 'react-router-dom';
2 import CabinsList_90 from '../components/CabinsList_90';
3 import { useCabins } from '../features/cabins/useCabins';
4 import Header_90 from '../components/Header_xx';
5
6 const Cabins_90 = () => {
7   const { cabins, isLoading } = useCabins();
8   return (
9     <>
10       <div className='antialiased bg-primary-950 text-primary-100 min-h-screen flex'>
11         <Header_90 />
12         <div className='flex-1 px-8 py-12 grid'>
13           <CabinsList_90 cabins={cabins} />
14         </div>
15       </div>
16     </>
17   );
18 };
19 export default Cabins_90;
20
```

=> 顯示與本題有關 useCabins() 及 services/apiCabins 相關的 code

```
code_90 > src > pages > JS Cabins_90.jsx > ...
1 import { Link } from 'react-router-dom';
2 import CabinsList_90 from '../components/CabinsList_90';
3 import { useCabins } from '../features/cabins/useCabins';
4 import Header_90 from '../components/Header_xx';
5
6 const Cabins_90 = () => {
7   const { cabins, isLoading } = useCabins();
8   return (
9     <>
10      <div className='antialiased bg-primary-950 text-primary-100 min-h-screen flex'>
11        <Header_90 />
12        <div className='flex-1 px-8 py-12 grid'>
13          <CabinsList_90 cabins={cabins} />
14        </div>
15      </div>
16    </>
17  );
18 };
19 export default Cabins_90;
20
```

```
code_90 > src > features > cabins > JS useCabins.js > useCabins
1 import { useQuery } from '@tanstack/react-query';
2 import { getCabins } from '../services/apiCabins';
3
4 export const useCabins = () => {
5   const {
6     data: cabins,
7     error,
8     isLoading,
9   } = useQuery({
10     queryKey: ['cabins-open-90'],
11     queryFn: getCabins,
12   });
13   return { cabins, error, isLoading };
14 };
15
```

```
code_90 > src > services > JS apiCabins.js > ...
1 import supabase, { supabaseUrl } from './supabase';
2
3 export async function getCabins() {
4   const { data: cabins, error } = await supabase.from('cabins_90').select('*');
5
6   if (error) {
7     console.error(error);
8     throw new Error('cabins could not be loaded');
9   }
10
11   console.log('getCabins()', cabins);
12
13   return cabins;
14 }
15
```

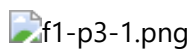
=> git log

```
git log --pretty=format:"%h%x09%an%x09%ad%x09%s" --after="2024-06-04"
0c9ffe3 sky001 Wed Jun 5 21:37:51 2024 +0800 upload final answer p1,p2,p3
```

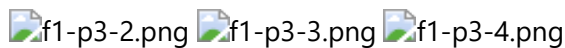
(25%) P3: 參考老師 w15 提供的 code，用路由 `/login_xx` 登入，用路由 `/logout_xx` 登出，登入帳號請以你在 Supabase 上已經建立有你學號的帳號。

老師這裡用的 Email 是 987654321@gmail.com, 密碼 13578642，你們必須要寫出本次考試登入時所用的帳號和密碼，方便老師或助教可以執行你們所寫的程式。

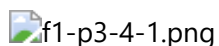
=> 登入帳號及密碼



=> 登入後，導向路由 `/guest/dashboard_xx`，呈現登入資訊如下三張圖

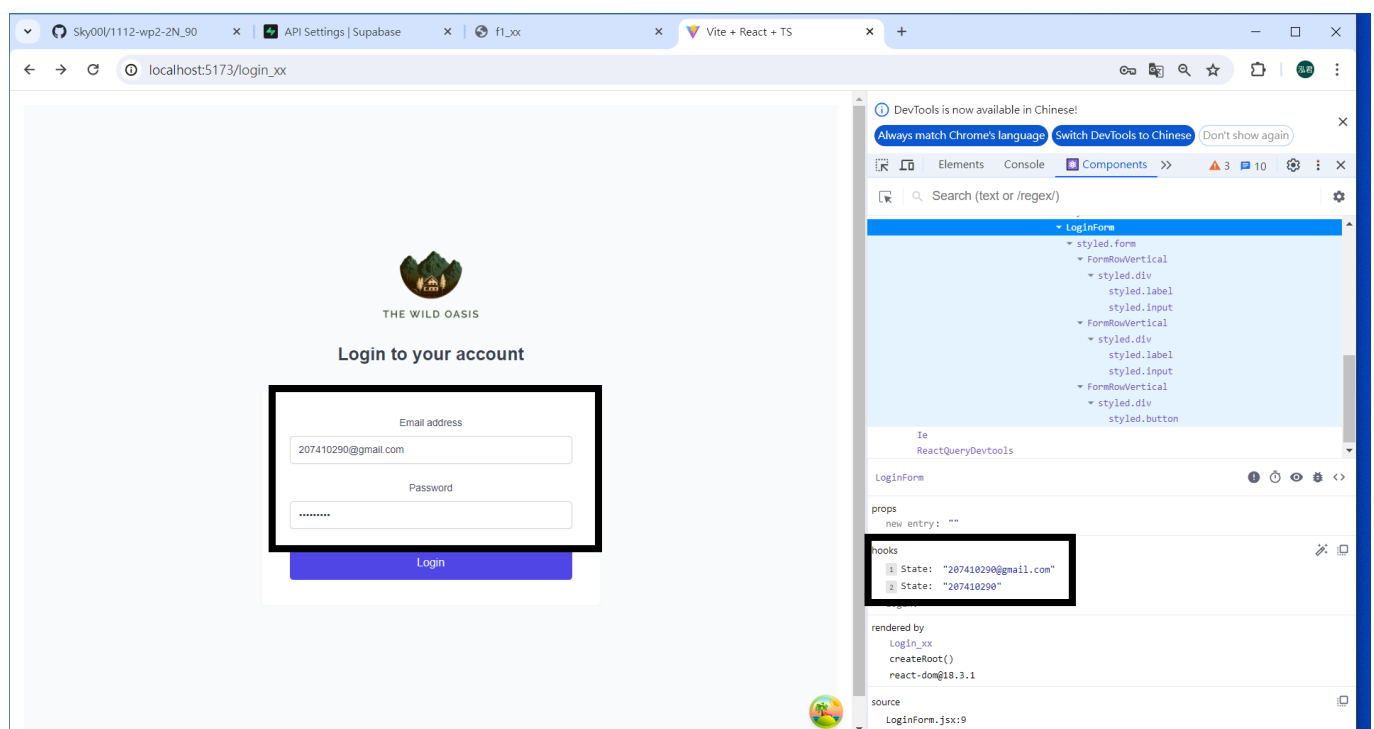


=> 目錄及檔案結構如下



Your Answer

=> 登入帳號及密碼



=> 登入後，導向路由 `/guest/dashboard_xx`，呈現登入資訊如下三張圖

Final Exam of Web prog2 2024

Guest name: 林泓君
Guest ID: 207410290
supabase email: 207410290@gmail.com
supabase userid: 9e30df4d-6968-46cf-a745-810d8719319b

supabase user info

project URL

API Settings

DevTools is now available in Chinese!

login() data

```
{
  "user": {
    "id": "9e30df4d-6968-46cf-a745-810d8719319b",
    "email": "207410290@gmail.com",
    "email_confirmed_at": "2024-06-04T14:59:57.471657Z",
    "created_at": "2024-06-04T14:59:57.467867Z",
    "last_sign_in_at": "2024-06-05T13:11:00.807686132Z",
    "phone": null,
    "role": "authenticated",
    "updated_at": "2024-06-05T13:11:00.8123412Z",
    "app_metadata": {
      "provider": "email",
      "providers": ["email"]
    },
    "identities": [
      {
        "provider": "email",
        "email": "207410290@gmail.com",
        "email_confirmed_at": "2024-06-04T14:59:57.471657Z",
        "created_at": "2024-06-04T14:59:57.467867Z",
        "last_sign_in_at": "2024-06-05T13:11:00.807686132Z",
        "phone": null,
        "role": "authenticated",
        "updated_at": "2024-06-05T13:11:00.8123412Z",
        "app_metadata": {
          "provider": "email",
          "providers": ["email"]
        }
      }
    ]
  },
  "session": {
    "access_token": "eyJhbGciOiJIUzI1NiIsImtpZCI6IjE6IiwiaWF0IjoiMTY3NzA0MDAwLjE6Iiwicm9va2kiOiJkaWVudCJ9",
    "expires_in": 3600,
    "refresh_token": "ARF8shoj2VG6FLhpZlaSZw",
    "token_type": "bearer"
  }
}
```

project URL

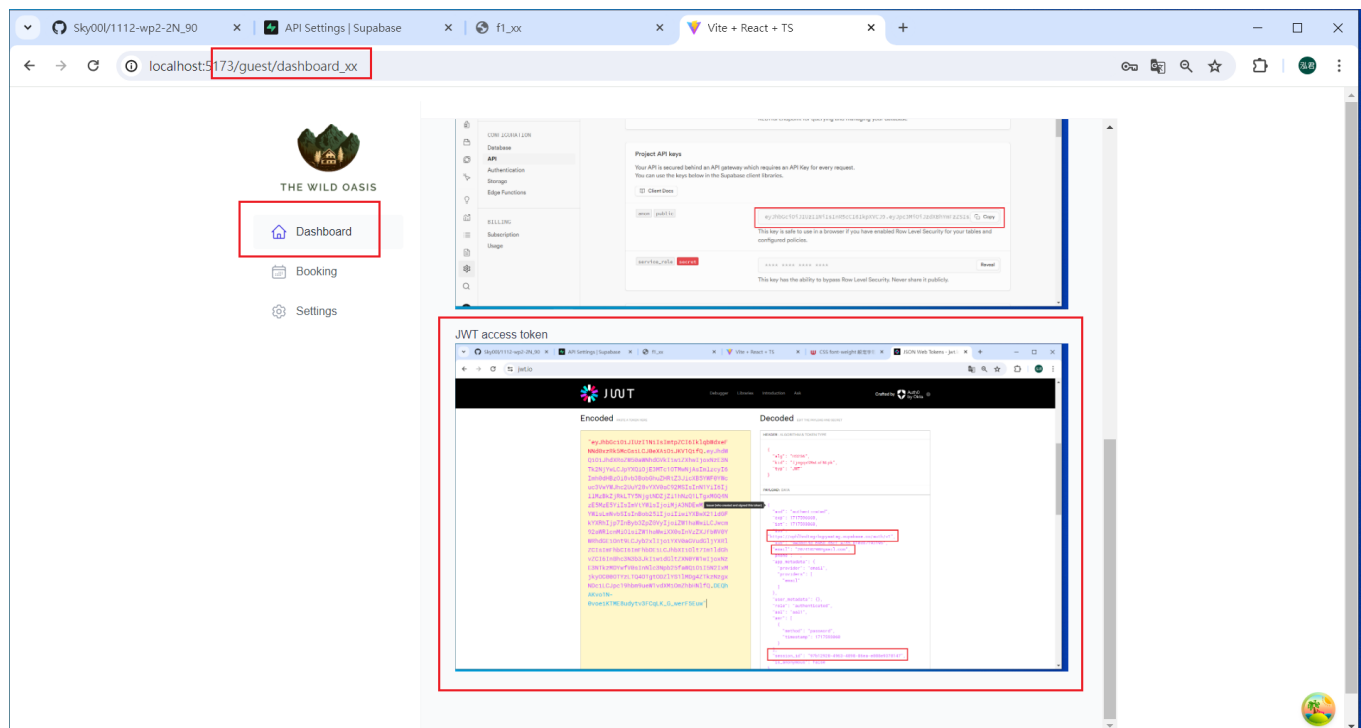
API Settings

JWT access token

Encoded

Decoded

```
{
  "header": {
    "alg": "HS256"
  },
  "payload": {
    "sub": "9e30df4d-6968-46cf-a745-810d8719319b",
    "email": "207410290@gmail.com",
    "email_confirmed_at": "2024-06-04T14:59:57.471657Z",
    "created_at": "2024-06-04T14:59:57.467867Z",
    "last_sign_in_at": "2024-06-05T13:11:00.807686132Z",
    "phone": null,
    "role": "authenticated",
    "updated_at": "2024-06-05T13:11:00.8123412Z",
    "app_metadata": {
      "provider": "email",
      "providers": ["email"]
    }
  },
  "signature": "eyJhbGciOiJIUzI1NiIsImtpZCI6IjE6IiwiaWF0IjoiMTY3NzA0MDAwLjE6Iiwicm9va2kiOiJkaWVudCJ9"
}
```



=> 顯示 guestDashboard_xx.jsx 相關 code

W16-final1-xx

code_90 > src > services > JS apiAuth.js

```
1 import supabase, { supabaseUrl } from './supabase';
2
3 export const login = async ({ email, password }) => {
4   console.log('login email password', email, password);
5   let { data, error } = await supabase.auth.signInWithPassword({
6     email,
7     password,
8   });
9   if (error) throw new Error(error.message);
10
11   console.log('login() data', data);
12   return data;
13 };
14
15 > export const getCurrentUser = async () => { ...
24 };
25
26 > export const logout = async () => { ...
29 };
30
```

code_90 > src > pages > JS Login_xx.jsx

```
1 import styled from 'styled-components';
2 import LoginForm from '../features/authentication/LoginForm';
3 import Logo from '../ui/Logo';
4 import Heading from '../ui/Heading';
5
6 const LoginLayout = styled.main`
7   min-height: 100vh;
8   display: grid;
9   grid-template-columns: 48rem;
10  align-content: center;
11  justify-content: center;
12  gap: 3.2rem;
13  background-color: var(--color-grey-50);
14 `;
15
16 function Login_xx() {
17   return (
18     <LoginLayout>
19       <Logo />
20       <Heading as="h4"> Login to your account</Heading>
21       <LoginForm />
22     </LoginLayout>
23   );
24 }
25
26 export default Login_xx;
27
```

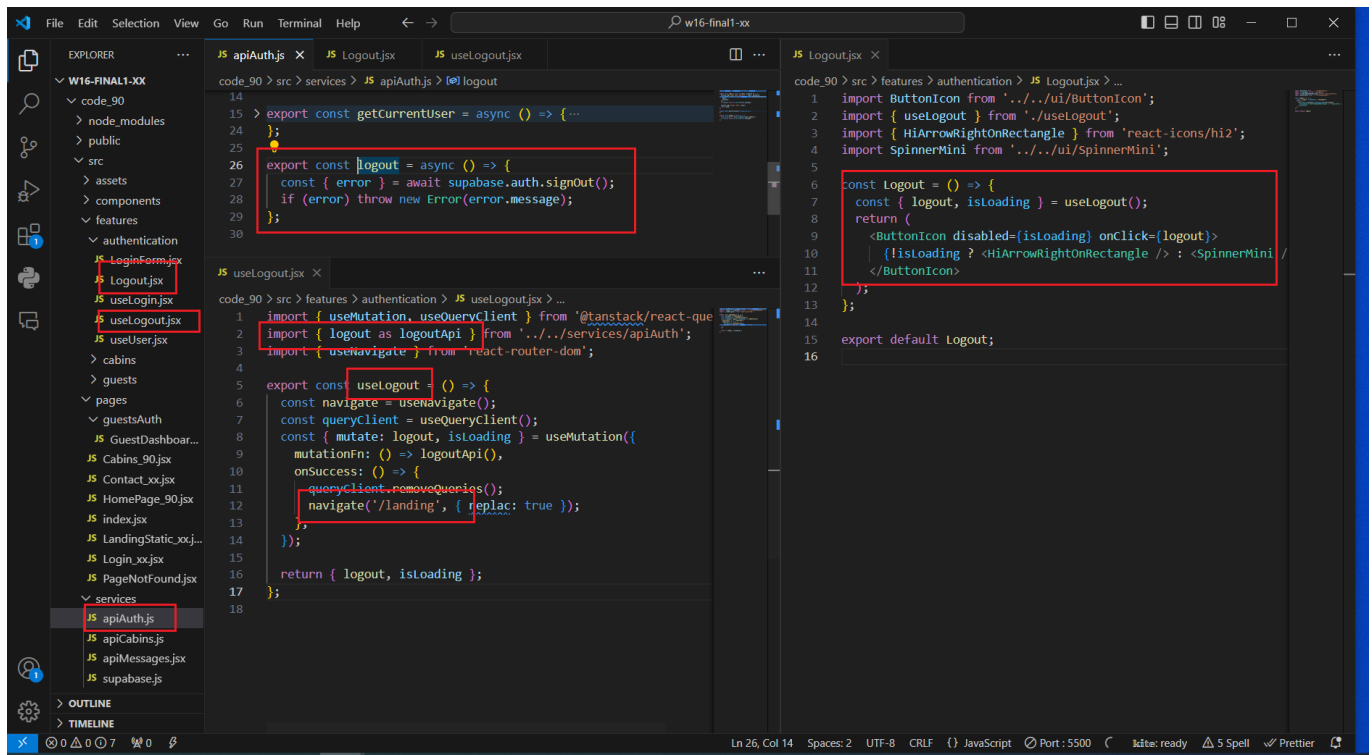
W16-final1-xx

code_90 > src > features > authentication > JS LoginForm.jsx

```
1 import { useState } from 'react';
2 import Button from '../ui/Button';
3 import Form from '../ui/Form';
4 import Input from '../ui/Input';
5 import FormRowVertical from '../ui/FormRowVertical';
6 import { useLogin } from './useLogin';
7
8 function LoginForm() {
9   const [email, setEmail] = useState('207410290@gmail.com');
10   const [password, setPassword] = useState('207410290');
11
12   const { login, isLoading } = useLogin();
13
14   function handleSubmit(e) {
15     e.preventDefault();
16     console.log('login form', email, password);
17     if (!email || !password) return;
18     login(
19       { email, password },
20       {
21         onSuccess: () => {
22           setEmail(''), setPassword('');
23         },
24       }
25     );
26   }
27
28   return (
29     <Form onSubmit={handleSubmit}>
30       <FormRowVertical label="Email address">
31         <Input
32           type="email"
33           id="email"
34           // This makes this form better for password managers
35           autoComplete="username"
36           value={email}
37           onChange={(e) => setEmail(e.target.value)}
38         />
39       </FormRowVertical>
40     </Form>
41   );
42 }
```

code_90 > src > features > authentication > JS useLogin.jsx

```
1 import { useMutation, useQueryClient } from '@tanstack/react-query';
2 import { login as loginApi } from '../services/apiAuth';
3 import { useNavigate } from 'react-router-dom';
4 import { toast } from 'react-hot-toast';
5
6 export const useLogin = () => {
7   const navigate = useNavigate();
8   const queryClient = useQueryClient();
9   const { mutate: login, isLoading } = useMutation({
10     mutationFn: ({ email, password }) => loginApi({ email, password }),
11     onSuccess: (data) => {
12       console.log('useLogin user', data);
13       queryClient.setQueryData(['user_xx'], data.user);
14       navigate('/guest/dashboard_xx', { replace: true });
15     },
16     onError: (err) => {
17       console.log('Error', err);
18       toast.error('provide email or password are incorrect');
19     },
20   });
21
22   return { login, isLoading };
23 };
24
```




=> git log

```
git log --pretty=format:"%h%x09%an%x09%ad%x09%s" --after="2024-06-04"
0c9ffe3 sky001 Wed Jun 5 21:37:51 2024 +0800 upload final answer p1,p2,p3
```


(30%) P4: 實作 /contact_xx，一開始可以新增一筆訊息到 Supabasa，取得一個 messageId 後，可以編輯。

=> 新增一筆訊息，按鈕是 Add Message，Supabase 端要有 messageId，可以透過呼叫 utils/helper.jsx 中的 createIdByDate() 取得。


 f1-p4-1.png

=> 新增一筆訊息回傳結果，按鈕變成 Edit Message


將 Room Reservation 改成 Room Reservation -- make change cabin 001 改成 cabin 003

 f1-p4-2.png

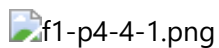
=> 可以看到已經改變

 f1-p4-3.png

=> 檢查 Supabase，也已經改變

 f1-p4-4.png

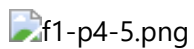
=> 檔案目錄結構如下



Your Answer

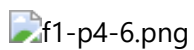
請根據上述圖片作答

=> 新增一筆訊息，按鈕是 **Add Message**

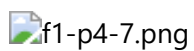


=> 新增一筆訊息回傳結果，按鈕變成 **Edit Message**

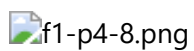
將 Room Reservation 改成 Room Reservation -- make change cabin 001 改成 cabin 003



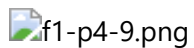
=> 可以看到已經改變



=> 檢查 **Supabase**，也已經改變



=> 顯示 **apiMessages.jsx code**



=> 顯示 **Contact.jsx 相關 code**



=> **git log**

