Back-end Questions

1. Explain First-party cookie & Third-party cookie

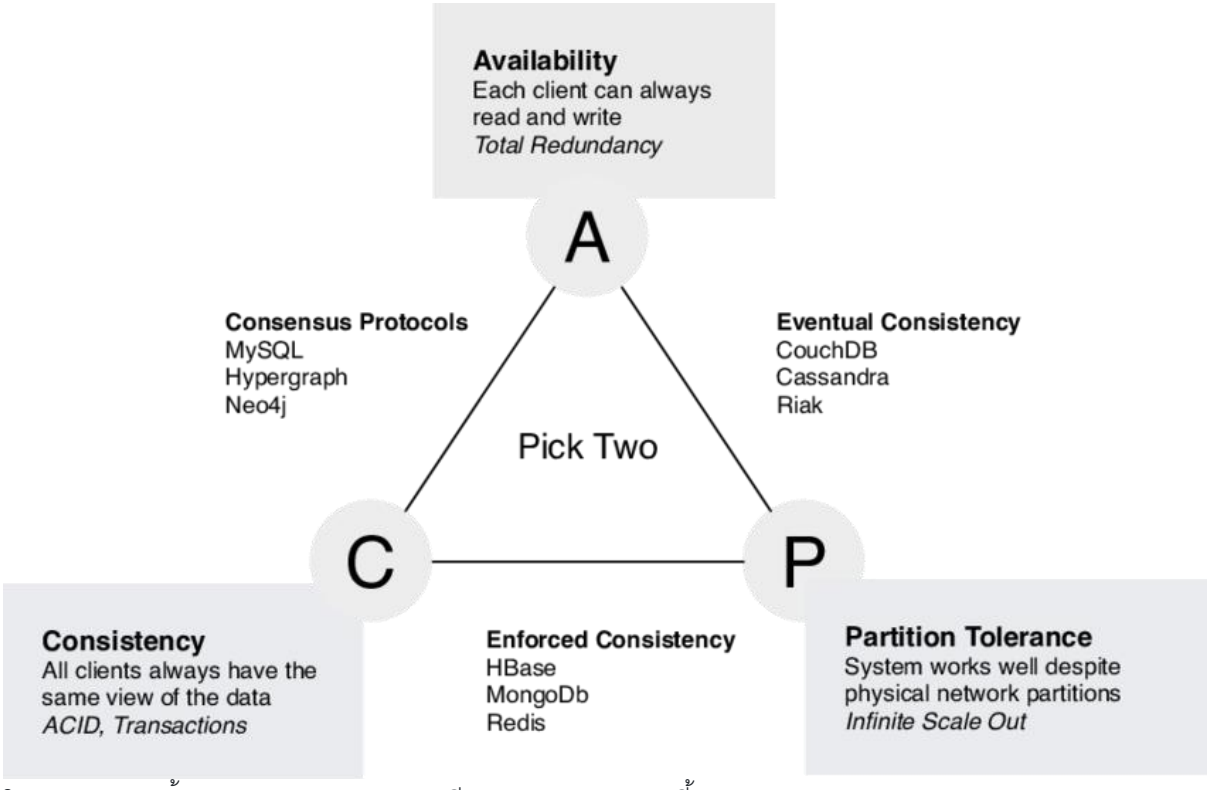
   Ans

   First-party Cookie เป็นคุกกี้ที่ทำหน้าที่เป็นความจำเกี่ยวกับข้อมูลของเว็บไซต์ที่เข้าใช้เพื่อให้เวลาเราเข้าเว็บไซต์ จะใช้ข้อมูลเดิมที่เราตั้งไว้หรืออื่นๆ เช่น username password

   Third-party Cookie เป็นคุกกี้ที่จะคล้ายกับด้านบนแต่ส่วนนี้จะมาจากเว็บไซต์อื่นๆที่เราไม่ได้เข้าไปใช้งานซึ่งหน้าที่หลักของคุกกี้ตัวนี้จะใช้ในการติดตามการเคลื่อนไหวในออนไลน์ของเราเพื่อนำข้อมูลเอาไปใช้งานในการวิเคราะห์ส่วนอื่นๆเพิ่มเติม

2. Explain CAP Theorem.

   Ans เป็นทฤษฎีนึงที่จะช่วยในการเลือก Database ที่ต้องการใช้ให้เหมาะสมกับงานที่เราต้องการ



โดยจากรูปภาพนั้น ความหมายแต่ละส่วนจะมีความหมายคร่าวๆดังนี้

Consistency = ข้อมูลที่เข้ามาเก็บภายใน Database จะต้องเป็นข้อมูลล่าสุดที่ถูกเก็บไว้เสมอ

Availability = ข้อมูลที่เราต้องการที่จะใช้นั้น พร้อมเสมอ

Partition Tolerance = ข้อมูลนั้นจะยังสามารถที่จะใช้งานได้อยู่แม้บางส่วนอาจจะเสียไป

3. Considering two queries

```
const searchIds = ['1', '2', '3', ...];
const query1 = await Product.find({ id: { $in: searchIds } });
const query2 = await Promise.all(searchIds.map(searchId =>
Product.find({ id: searchId })));
```

Which one is faster.

Ans query2 เพราะเนื่องจาก ข้อมูลที่ได้ออกมานั้นจะรวมออกมาเป็นชุดเดียว

4. Explain XSS / SQL Injection / Man in the Middle Attack, and how to prevent each attack type.

Ans

Xss เป็นการส่งหรือฝัง script เข้าไปในหน้าเว็ป เมื่อผู้ใช้ทำการโหลดหน้าเว็บ script ก็จะทำงาน เพื่อเอาข้อมูลสำคัญและเมื่อผู้ใช้มีการโหลดหน้าเว็บก็ให้ทำการส่งข้อมูลสำคัญให้กับผู้ที่ต้องการ

วิธีป้องกัน

- เลือกใช้ framework ในการพัฒนา website
- ทำการ encode ข้อมูลต่างๆ ที่ส่งออกไป
- ใช้ Content Security Policy เพื่อตรวจสอบว่าได้มาจากไหนหรือจากแหล่งที่ถูกต้อง

SQL Injection เป็นการส่งข้อมูลเข้าไปเพื่อต้องการโจมตีฐานข้อมูลจากทาง from ต่างๆ

วิธีป้องกัน

- ตรวจสอบ input ที่ส่งเข้ามาว่าถูกต้องตามที่เราต้องการหรือไม่
- ใช้อักษรพิเศษในการกรองข้อมูลเพื่อส่งข้อมูลเข้าสู่ database

Man in the Middle Attack เป็นการที่ผู้โจมตีปลอมตัวเข้ามาเป็นตัวกลางในการรับส่งข้อมูลระหว่างผู้ใช้และเซิฟเวอร์

วิธีป้องกัน

- การใช้ SSL หรือใบรับรองเว็บไซต์เพื่อป้องกัน
- การใช้โครงสร้างของ Internet ต่างๆที่จะใช้ส่งข้อมูลสำคัญ เช่น การเงิน ในกรณีของ บริษัทต่างๆก็จะใช้ network ในการแยกส่งเช่นการผ่าน VPN ต่างๆ ออกไปอีกขั้นนึง

5. Explain the different between using callback / Promise / async await. When to use and when not to.

Ans

Callback คือ function ถูกเรียกใช้งานหลังจากที่เราใช้ function ที่เราใช้เสร็จ เพื่อให้ทำงานเป็น synchronous ซึ่งถ้าหาก Callback มีเยอะมากๆจะทำให้เกิด Callback hell ซึ่งทำให้เราตรวจสอบ error ยากและทำให้ code นั้นอ่านยากมากๆ

Promise เป็นการรอ function ก่อนหน้าที่เราใช้งานเพื่อ confirm ว่า function ที่เรานั้นต้องการให้ทำงานต่อ จะต้องรอ function นี้อย่างแน่นอน และการใช้ Promise เป็นการเอามาแก้ปัญหา callback hell แต่ก็ยังมีปัญหาเรื่องของการดักจับ error อยู่

async / await ทำหน้าที่เหมือนกับ Promise แต่เอามาใช้เพื่อให้อ่านสะดวกยิ่งขึ้นมากว่า Promise แถมยังดักจับ error ได้ง่ายมากขึ้น

การใช้ในแต่ละสถานการณ์ ของแต่ละอย่าง

callback ควรเอามาใช้ก็ต่อเมื่อ function นั้นมีการใช้บ่อยๆหรือต้องการใช้ซ้ำ

promise / async / await ควรเอามาใช้ในตอนที่เราต้องการให้ function เหล่านั้นรอหรือมีดารทำงานที่เสร็จพร้อมกันจริงๆ หรือบางครั้ง เราออาจจะใช้ร่วมกันในบางกรณี อย่างเช่น เราต้องการที่จะให้ API 2 อย่างเสร็จพร้อมๆกัน

6. Explain how HTTP protocol works.

- เริ่มต้นด้วยการเชื่อมต่อระหว่าง Browser ของผู้ใช้และเซิฟเวอร์
- เมื่อทำการเชื่อมต่อเรียบร้อย Browser ก็จะ request ข้อมูลจากเซิฟเวอร์
- เซิฟเวอร์ส่งข้อมูลให้กับ Browser ตามที่ request มา
- Browser นำข้อมูลมาแสดงให้กับทางผู้ใช้