11.

ตอบ. 1.มีการเก็บCodeเก่าๆไว้ไม่ต้อง สำเนาเอง

2.มีการบอกว่าเพิ่ม ลบ อะไรเข้ามา

3.มีTimeline บอกว่าวันไหนทำงานเยอะหรือน้อย

4.มี Graph บอกว่าเราเขียนภาษาอะไรเยอะแค่ไหน

12.

ตอบ. ไฟล์ทั้งหมดไม่ได้ถูกเก็บอยู่ที่ Central repo หรือศูนย์กลางที่เดียวทำให้ เวลา Central repo เกิดพังขึ้นมา ผู้ใช้ก็ยังสามารถดึงไฟล์งานออกมาได้

13

ตอบ. ไฟล์ถูกเก็บรักษาในที่เดียวทำให้ง่ายต่อการดูแลรักษา

14

ตอบ. เมื่อมีการMergeแล้วเกิดการ Conflict จะมีบรรทัดที่มีการ เออเร่อมาให้

15

ตอบ.

16

ตอบ. Git คือ Version Control ตัวหนึ่ง ซึ่งเป็นระบบที่มีหน้าที่ในการจัดเก็บการเปลี่ยนแปลงของไฟล์ในโปรเจ็คเรา มีการ backup code ให้เรา สามารถที่จะเรียกดูหรือย้อนกลับไปดูเวอร์ชั่นต่างๆของโปรเจ็คที่ใด เวลาใดก็ได้ หรือแม้แต่ดูว่าไฟล์นั้นๆใครเป็นคนเพิ่มหรือแก้ไข หรือว่าจะดูว่าไฟล์นั้นๆถูกเขียนโดยใครบ้างก็สามารถทำได้ ฉะนั้น Version Control ก็เหมาะอย่างยิ่งสำหรับนักพัฒนาไม่ว่าจะเป็นคนเดียวโดยเฉพาะอย่างยิ่งจะมีประสิทธิภาพมากหากเป็นการพัฒนาเป็นทีม

แตกต่างจาก Gibhub เพราะGIbhub คือเว็บเซิฟเวอร์ ที่ให้บริการในการฝากไฟล์ git คือตัวที่เราต้องการนำไปฝาก โหลดมาได้แต่แก้ไขโค้ดใน git ไม่ได้

17

ตอบ. เพื่อทำความเข้าใจวิธีที่ Git แตก branch เราต้องย้อนกลับมาดูว่า Git เก็บข้อมูลยังไง ตามที่คุณอาจจะจำได้จาก Chapter 1 ว่า Git ไม่ได้เก็บข้อมูลเป็นลำดับของความเปลี่ยนแปลงต่อเวลาแต่เก็บเป็นลำดับของ snapshot ต่อเวลา

เวลาคุณ commit ใน Git นั้น Git จะเก็บเป็น object ของการ commit ซึ่งประกอบด้วย pointer ชี้ไปยัง snapshot ของ content ที่คุณ stage ไว้, metadata ของชื่อผู้แก้ไขและ message ที่บันทึกไว้ และ pointer ที่ชี้ไปยัง parent ลำดับถัดไปของ commit นั้นๆ (ซึ่งอาจจะไม่มีก็ได้ถ้าเป็น commit ครั้งแรก, อาจจะมีอันเดียวชี้ไปยัง parent ของ commit ปรกติทั่วไปหรืออาจจะมี parent หลายอันสำหรับ commit ที่เป็นผลจากการ merge หลายๆ branch เข้าด้วยกัน)

18

ตอบ

19

ตอบ. คือการรวมโค๊ดจาก remote มายัง local โดยที่เราไม่สามารถรู้ได้เลยว่าจะรวมโค๊ดอะไรบ้าง รู้แค่หลังจาก pull เสร็จแล้ว ซึ่งจริงๆแล้ว git pull มันคือการทำ git fetch และต่อด้วย git merge อัตโนมัติ

20

ตอบ. Git branch model เป็นกระบวนการแยก branch ออกจากmaster และ merge กลับเข้าภายหลัง