git\_spurningar.txt

1. **Hvað gera eftirfarandi Linux skipanir?**

***Cd:*** Breyta Directory.

***Cd ~:*** þetta gerir notandanum að skila strax eða beint til heimasvæði sínu

***ls:*** þetta List upplýsingar um skrá

***pwd:*** þetta framköllun Vinna Directory

***mkdir:*** Þetta skapar nýja möppu

2. **Hvað gera eftirfarandi git skipanir og hvers vegna eru þær gagnlegar?:**

git clone: það skapar fjarlægur-rekja útibú fyrir hverja grein í klóna geymsla (sýnilegt með Git útibú R), og skapar og tékka út fyrstu útibú sem er Forked úr virku útibúi klóna geymsla, er í raun eitthvað af a. umbúðir í kring nokkrum öðrum skipunum, Clones geymsla í nýstofnaða skrá, þess er notað til að skapa nýja möppu, fer inn í það og keyrir

git log: það sýna náðist skráð sögu verkefni sem nýjasta fremja skyndimynd aftur á bak. Við vanræksla það mun aðeins sýna sögu útibúsins þú ert núna á, en hægt er að gefa mismunandi eða jafnvel margar höfuð eða útibú sem yfirferðar. Það er oft notað til að sýna mismun milli tveggja eða fleiri greinum á the fremja stigi.

git status: þetta sýnir mismunandi ríkjum skrá í vinnumöppu og sviðsetning svæði. Hvaða skrár eru breytt og unstaged og sem eru sett á svið, en ekki enn framið. Í venjulegum formi, það er notað til að sýna sumir undirstöðu vísbendingar um hvernig á að flytja skrár á milli þessara stigum.

git diff: þetta er og notað til að sjá muninn á milli tveggja trjáa. Þetta gæti verið munur á milli aðbúnað og sviðsetning svæði.

git checkout: þetta er og er notað til að skipta útibú og athugaðu efni út í vinnumöppu þitt

3. **Hver er munurinn á eftirfarandi git skipunum:**

a) git diff : er munur þegar þú vilt sjá muninn hvaða tvö tré.

b) git diff –staged:á meðan þetta er ef þú vilt sjá hvað þú hefur sett það mun fara inn í þinn næst fremja, þú getur notað þessa skipun samanburð leiksvið breytingarnar til síðasta þína fremja

c) git diff commit1 commit2: Commit1 skráir fyrst fremja mynd sem þú settir upp í stigun þínu svæði meðan commit2 er annar skuldbinda sig til að bæta við sögu.

4. **Hvað er útgáfustýring (e.version control)?:**

Það er kerfi sem skráir breytingar á skrá eða setja af skrá yfir tíma þannig að þú getur muna ákveðin útgáfur síðar

5. **Hverjir eru helstu kostir við að nota GIT?:**auðveldlega uppgötva. Git er mjög hratt, jafnvel þegar miðað er við önnur DVCS, fyrir staðbundin auk netrekstri

Free: Ég er viss um að enginn vill eyða $ 450 fyrir persónulega verkefni þeirra. framkvæmdastjóri mun meta það ef þú bjarga honum N x $ 450

6. **Hversu oft telur þú að eigi að gera commit í verkefni, rökstuddu?:**

Ég held ekki að þú ættir að hafa áhyggjur svo mikið um hversu oft. The mikilvægur hlutur hér er hvað, hvenær og hvers vegna. Commit þegar þú hefur eitthvað að fremja, ekki ef þú gerir það ekki.

[...] Ef þú ert að gera margar breytingar á verkefni á sama tíma, skipta þeim upp í rökrétt hluta og fremja þá í mörgum fundum. Þetta gerir það mun auðveldara að rekja sögu einstakra breytinga, sem mun spara þér mikinn tíma þegar reynt var að finna og laga galla síðar. Til dæmis, ef þú ert að innleiða lögun a, b og C og ákveða galla 1, 2 og 3, sem ætti að leiða í samtals amk sex skuldbindur, einn fyrir hverja aðgerð og einn fyrir hvern galla. Ef þú ert að vinna á stóru lögun eða gera víðtækur refactoring, íhuga að skipta vinnu þinni upp í jafnvel smærri hluta, og gera fremja eftir hverjum hluta er lokið. Einnig, þegar innleiða sjálfstæða breytingar á mörgum rökrétt mát, fremja breytingar hverja einingu fyrir sig, jafnvel þótt þeir séu hluti af stærri breytingu.

Fullkomlega, ættir þú aldrei yfirgefa skrifstofu með inngang breytingum á harða diskinum. Ef þú ert að vinna á verkefnum þar sem breytingar munu hafa áhrif öðru fólki, íhuga að nota útibú til að innleiða breytingar og sameina þá aftur í skottinu þegar þú ert búinn. Þegar fremja breytingar á bókasöfnum eða verkefnum sem önnur verkefni-og þannig, annað fólk-reiða sig á, að tryggja að þú brjóta ekki þeirra byggir því að fremja kóða sem mun ekki taka saman. Hins vegar hafa kóða sem gerir ekki saman er ekki afsökun til að forðast að fremja. Nota útibú í staðinn. [...]

7.**Hvað er "Working directory"?" "Staging area"?" og "Repository" í GIT?:**

"Working directory" er einn stöðva af einni útgáfu af verkefninu. Þessar skrár eru dreginn út af samþjöppuðum gagnagrunninum í Git skrá og sett á disk fyrir þig að nota eða breyta.

"Staging area" er skrá, venjulega að finna í Git skrá, sem geymir upplýsingar um hvað megi fara inn um næstu þitt fremja. Það er stundum vísað til sem "vísitölu", en það er líka algengt að vísa til þess sem sviðsetning svæði.

Og "Repository" í GIT er þar Git geymir lýsigögn og mótmæla gagnagrunnur fyrir verkefnið þitt. Þetta er mikilvægur hluti af Git, og það er það sem er afrituð þegar klón geymsla frá annarri tölvu.

8.**Hvenær er sniðugt að nota greinar(branches)?:**

9.**Hver fann upp GIT?:** Linus Torvalds**.**

10.**Hvar er helst notað GIT ?:**

Þetta er greiðslumáti eftirnafn fyrir ecommerce vettvang Magento. Nota það helst ef þú viðhalda búð fyrir læknisfræðilegum hluti þar sem það ætti að vera hægt að greiða í gegnum lyfseðils.

Það gerir búð eigandi að láta viðskiptavini að velja hvaða atriði skal greidd með lyfseðils. Þeir hlutir gildi verður að vera stillt á 0,00. Öll önnur atriði er hægt að greiða með öðrum greiðslumáta (f. Ex. PayPal).