



**University of Ibb
Faculty of Science
Department of Information Technology**

**Level: Fourth
Course: Software Engineering
Instructor: Eng. Malek Al-Munsif**

**Student Name:
*Abdulrahman Moti'a Mohammed***

**Lecture Implementation:
lec01 - Introduction**



MINGW64:/c/Users/abdua/localization-styles-app/lec01-intro

```
Abdulrhman@Abdulrhman-Sinan MINGW64 ~
$ git config --global user.name "Abdulrhman-Sinan"
git config --global user.email "bdalrhmansinan1@example.com"

Abdulrhman@Abdulrhman-Sinan MINGW64 ~
$ git config --global user.name
Abdulrhman-Sinan

Abdulrhman@Abdulrhman-Sinan MINGW64 ~
$ git config --global user.email
bdalrhmansinan1@example.com

Abdulrhman@Abdulrhman-Sinan MINGW64 ~
$ git config --global --list
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %F
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
user.name=Abdulrhman-Sinan
user.email=bdalrhmansinan1@example.com
core.autocrlf=true
```

الأوامر المستخدمة لتهيئة Git:

- "git config --global user.name "Abdulrhman-Sinan" •
هذا الأمر يستخدم لتحديد اسم المستخدم الذي سيظهر في سجل التعديلات (commits) لجميع المشاريع على الجهاز.
- "git config --global user.email "bdalrhmansinan1@example.com" •
يستخدم هذا الأمر لتحديد البريد الإلكتروني الذي سيرتبط أيضًا بكل تعديل تقوم به على Git.
- git config --global user.name •
يعرض الاسم المخزن حالياً كقيمة لـ user.name.
- git config --global user.email •
يعرض البريد الإلكتروني المخزن حالياً كقيمة لـ user.email.
- git config --global --list •
يعرض جميع الإعدادات العامة (global) التي تم تهيئتها في Git، بما في ذلك الاسم، البريد الإلكتروني، وأي إعدادات أخرى مثل التعامل مع الملفات الكبيرة.

```

Abdulrhman@Abdulrhman-Sinan MINGW64 ~
$ git clone https://github.com/abdulrhmanmotia/localization-styles-app.git
cd localization-styles-app
Cloning into 'localization-styles-app'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app (main)
$ mkdir lec01-intro
cd lec01-intro
echo "# المحاضرة الأولى - مقدمة" > README.md
cd ..

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app (main)
$ ls -R
.:
README.md  lec01-intro/

./lec01-intro:
README.md

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app (main)
$ cd ~/localization-styles-app/lec01-intro

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ cat > README.md << 'EOF'
@ d ~/localization-styles-app/lec01-intro
at > README.md << 'EOF'
@ d ~/localization-styles-app/lec01-intro
at > README.md << 'EOF'
@ d ~/localization-styles-app/lec01-intro
at > README.md << 'EOF'

```

ما تم تنفيذه:

في هذا الجزء، قمنا بتنفيذ مجموعة أوامر Git لإنشاء مجلد خاص بالمحاضرة الأولى ضمن مشروع موجود على GitHub. تضمنت الخطوات ما يلي:

1. استنساخ المشروع من GitHub باستخدام `git clone` لتوفير نسخة محلية.
2. الدخول إلى مجلد المشروع المستنسخ.
3. إنشاء مجلد جديد باسم `lec01-intro` مخصص للمحاضرة الأولى.
4. إضافة ملف `README.md` يحتوي على عنوان المحاضرة.
5. عرض هيكل المجلدات والملفات باستخدام `ls` للتأكد من وجود الملف داخل المجلد.
6. إعادة تعديل محتوى ملف `README.md` باستخدام أمر `cat > README.md` لإضافة وصف أو محتوى جديد بشكل مباشر.

```

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git add README.md
git commit -m "إضافة ملف README لمحاضرة Git الأساسية"
warning: in the working copy of 'Tec01-intro/README.md', LF will be replaced by CRLF the next time Git
touches it
[main ecbfbb1] [ملف README لمحاضرة Git الأساسية]
  1 file changed, 85 insertions(+)
   create mode 100644 tec01-intro/README.md

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 1.88 KiB | 1.88 MiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/abdulrhmannotia/localization-styles-app.git
  e659129..ecbfbb1 main -> main

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ cat README.md
كما وفتحنا بيئة العمل Git في هذه المحاضرة تعلمنا أساسيات التعامل مع نظام التحكم في إصدارات
ات المشروع وتتبع التغييرات.

git init

```

الأوامر التينفذت:

1. git add README.md

أضفت ملف README.md إلى منطقة التحضير (staging area) ليتم تضمينه في عملية الحفظ (commit).

2. git commit -m "إضافة ملف README لمحاضرة Git الأساسية"

حفظت التغييرات في المستودع المحلي مع رسالة توضيحية تُظهر أنك أضفت ملف README لمحاضرة Git الأساسية.

a. التحذير: LF يعني أن نظام Git سيحول نهاية السطور من صيغة Windows (CRLF) إلى صيغة Unix (LF) عند التعامل مع الملف لاحقاً.

3. git push origin main

أرسلت (رفعت) التغييرات التي قمت بحفظها (commit) من المستودع المحلي إلى المستودع البعيد main على الفرع GitHub.

4. cat README.md

عرض محتويات ملف README.md مباشرة في الطرفية.

```

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git init
Initialized empty Git repository in C:/Users/abdua/localization-styles-app/lec01-intro/.git/

```

أمر git init

من تتبع التغييرات التي ستحدث داخل Git جديد فارغ داخل المجلد الحالي، ليتمكن Git يقوم هذا الأمر بإنشاء مستودع هذا المجلد.

يحتوي على جميع ملفات التحكم بالإصدار الخاصة بالمشروع git. بعد تنفيذه، ستجد مجلداً مخفياً اسمه .git.

```
Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git status
On branch main
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
```

git status

يعرض هذا الأمر حالة المستودع الحالي، ويخبرك عن:

- الفرع الذي تعمل عليه حالياً (في المثال: main).
- هل هناك تغييرات لم تُرتكب (لم يتم commit) بعد في ملفات المشروع.
- إذا كانت نسخة الفرع المحلية متزامنة مع النسخة الموجودة على المستودع البعيد (remote).

```
Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git add .

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git commit -m "first commit"
On branch main
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git remote add origin https://github.com/abdulrhmanotia/localization-styles-app.git
error: remote origin already exists.

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git push -u origin master
```

git add .

يضيف كل التغييرات الحالية (ملفات جديدة أو معدلة) إلى منطقة الاستعداد (staging area) عشان تُحفظ في commit.

m "first commit"- git commit

يحفظ التغييرات المضافة في سجل Git مع رسالة توضح سبب التعديل.

الرسالة "working tree clean, nothing to commit" تعني ما في تغييرات جديدة للحفظ.

git remote add origin <url>

يربط المستودع المحلي بالمستودع البعيد على GitHub باسم "origin".

رسالة الخطأ "remote origin already exists" تعني أنه تم الربط من قبل ولا يمكن إعادة الربط بنفس الاسم.

u origin master- git push

- يدفع التغييرات إلى الفرع "master" على المستودع البعيد ويربط الفرع المحلي مع البعيد.
- رسالة الخطأ "src refspec master does not match any" تعني أن الفرع "master" غير موجود محلياً، ربما تستخدم فرع "main" بدلاً منه.

```
Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git branch -M main

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git pull origin main
remote: Enumerating objects: 7, done.
remote: Counting objects: 100% (7/7), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 7 (delta 0), reused 4 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (7/7), 2.73 KiB | 159.00 KiB/s, done.
From https://github.com/abdulrhmannotia/localization-styles-app
 * branch            main      -> FETCH_HEAD
 + be3d11e...ecfbff1 main      -> origin/main (forced update)
fatal: refusing to merge unrelated histories

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git log
commit 7165e1832fc26327dce93bd17f5b8a6559991f1d (HEAD -> main, origin/master)
Author: Abdulrhman-Sinan <bdalrhmansinan1@example.com>
Date:   Wed Jul 30 22:40:44 2025 +0300

    first commit

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
echo "هذا محتوى الملف التجريبى" > example.txt

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ ls -l
total 5
-rw-r--r-- 1 Abdulrhman 197121 4028 Jul 30 22:33 README.md
-rw-r--r-- 1 Abdulrhman 197121    46 Jul 30 22:54 example.txt
```

M main- git branch .1

- تغيير اسم الفرع الحالي إلى "main" بدل اسم الفرع الافتراضي من master إلى main.

git pull origin main .2

- جلب التحديثات من الفرع "main" في المستودع البعيد ومحاولة دمجها مع الفرع المحلي لكن ظهرت مشكلة لأن المستودع المحلي والبعيد لهما تاريخ مختلف (unrelated histories).).

git log .3

- عرض سجل الالتزامات (commits) في المستودع المحلي يعرض قائمة التغييرات التي تمت مع معلومات عن كل التزام.

example.txt < echo .4

- إنشاء ملف نصي جديد باسم example.txt وكتابة نص بداخله أنشأت ملف تجاري للمشروع.

1- ls .5

- عرض محتويات المجلد الحالي بتفاصيل (تفاصيل الأذونات، الحجم، التاريخ، اسم الملف)
للتأكد من وجود الملفات مثل README.md و example.txt

```
Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git add example.txt
warning: in the working copy of 'example.txt', LF will be replaced by CRLF the next time Git touches it

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git status
On branch main
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   example.txt

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ touch test-file.txt
git add test-file.txt

Abdulrhman@Abdulrhman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ cat << 'EOF' > demo.txt

EOF
git add demo.txt
warning: in the working copy of 'demo.txt', LF will be replaced by CRLF the next time Git touches it
```

git add example.txt .1

- أضفت الملف **example.txt** إلى منطقة التهيئة (staging area) استعداداً لتسجيله في commit.
- ظهر تحذير بأن نوع نهاية السطر (LF) سيتم تغييره إلى CRLF عند التعامل معه على ويندوز، وهو أمر شائع ولا يؤثر غالباً.

git status .2

- عرضت حالة المستودع الحالية، وعرفت أن الفرع الحالي هو **main**، وهو متزامن مع الفرع البعيد **origin/master**.
- أظهر الأمر أن **example.txt** مضاد وجاهز للتسجيل (staged).

touch test-file.txt .3

- أنشأت ملف فارغ جديد باسم **test-file.txt** في المجلد الحالي.

git add test-file.txt .4

- أضفت الملف الجديد **test-file.txt** إلى منطقة التهيئة (staging area) استعداداً لتسجيله.

```

bash
cat << 'EOF' > demo.txt
السطر الأول
السطر الثاني
هذا محتوى تجريبى
EOF

```

- أنشأت ملف نصي جديد اسمه **demo.txt** يحتوي على 3 أسطر نصية محددة داخل الأمر.
- هذه طريقة سريعة لإنشاء ملف بمحنتى معين من خلال الطرفية.

git add demo.txt .6

- أضفت الملف **demo.txt** إلى منطقة التهيئـة (staging area) ليتم تضمينه في commit لاحقاً.

ملخص عام:

قمت بإنشاء 3 ملفات جديدة (**demo.txt**, **test-file.txt**, **example.txt**)، وأضفتها كلها إلى منطقة التهيئـة (staging area) استعداداً (staging area) استعداداً، وأضفتها كلها إلى منطقة التهيئـة (staging area). أمر **git status** (commit) استخدمته للتحقق من حالة الملفات في المستودع لتسجيلها في المستودع (commit). أمر **git status** (commit) استخدمته للتحقق من حالة الملفات في المستودع لتسجيلها في المستودع (commit).

```

Abdu1rhaman@Abdu1rhaman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ pwd # لتأكيد من معرفة المكان
/c/Users/abdua/localiztion-styles-app/lec01-intro
Abdu1rhaman@Abdu1rhaman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ echo "اولاً ملف تجربى للتعلم" > test-file.txt
Abdu1rhaman@Abdu1rhaman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git add test-file.txt
warning: in the working copy of 'test-file.txt', LF will be replaced by CRLF the next time Git touches it
Abdu1rhaman@Abdu1rhaman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git log --oneline
7165e18 (HEAD -> main, origin/master) first commit
Abdu1rhaman@Abdu1rhaman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git branch new-feature
Abdu1rhaman@Abdu1rhaman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git checkout new-feature
A demo.txt
A example.txt
A test-file.txt
Switched to branch 'new-feature'
Abdu1rhaman@Abdu1rhaman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (new-feature)
$ git checkout -b hotfix
Switched to a new branch 'hotfix'
Abdu1rhaman@Abdu1rhaman-Sinan MINGW64 ~/localiztion-styles-app/lec01-intro (hotfix)
$ git checkout main
git merge hotfix
A demo.txt
A example.txt
A test-file.txt
Switched to branch 'main'
Your branch is up to date with 'origin/master'.
Already up to date.
Abdu1rhaman@Abdu1rhaman-Sinan MINGW64 ~/localization-styles-app/lec01-intro (main)
$ git push -u origin hotfix

```

pwd .1

- يعرض المسار الكامل للمجلد الحالي الذي تعمل فيه (التأكد من موقعك في النظام).

echo .2 "هذا ملف تجريبى للتعلم" < test-file.txt

- ينشئ ملفاً جديداً باسم **test-file.txt** ويكتب فيه النص "هذا ملف تجريبى للتعلم".
- إذا كان الملف موجوداً، سيتم استبدال محتواه.

git add test-file.txt .3

- يضيف الملف **test-file.txt** إلى منطقة التبيئة (staging area) ليتم تضمينه في commit لاحقاً.
- ظهر تحذير متعلق بتغيير نوع نهاية السطر (CRLF إلى LF)، وهو أمر شائع على أنظمة ويندوز.

git log --oneline .4

- يعرض سجل التعديلات (commits) بشكل مختصر (سطر واحد لكل commit).
- من خلاله رأيت أن هناك commit واحد بعنوان "first commit" والفرع الحالي **main** متزامن مع الفرع البعيد **origin/master**.

git branch new-feature .5

- أنشأ فرع جديد باسم **new-feature** في المستودع، ولكنه لم ينتقل إليه بعد.

git checkout new-feature .6

- انتقلت إلى الفرع الجديد **new-feature**.
- أظهر Git أن الملفات الثلاثة (test-file.txt ,example.txt ,demo.txt) تم وضعها في منطقة الإضافة (staged) ولم يتم عمل commit لهم بعد.

b hotfix- git checkout .7

- أنشأ فرع جديد باسم **hotfix** وانتقلت إليه مباشرةً.

- هذا اختصار لإنشاء فرع جديد والانتقال إليه في خطوة واحدة.

git checkout main .8

- رجعت إلى الفرع **main**.

git merge hotfix .9

- مجت التعديلات الموجودة في الفرع **hotfix** إلى الفرع **main**.
- لكن ظهرت رسالة "Already up to date". مما يعني أنه لا يوجد تغييرات جديدة في فرع **hotfix** غير موجودة في **main**.

```
AbduLrhman@AbduLrhman-Sinan MINGW64 ~/localiztion-styles-app/lec01-intro (main)
$ git push -u origin hotfix
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 1.70 KiB | 1.71 MiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'hotfix' on GitHub by visiting:
remote:   https://github.com/abduLrhmanmotia/localiztion-styles-app/pull/new/hotfix
remote:
To https://github.com/abduLrhmanmotia/localiztion-styles-app.git
 * [new branch]      hotfix -> hotfix
branch 'hotfix' set up to track 'origin/hotfix'.

AbduLrhman@AbduLrhman-Sinan MINGW64 ~/localiztion-styles-app/lec01-intro (main)
$ git pull origin main
From https://github.com/abduLrhmanmotia/localiztion-styles-app
 * branch            main      -> FETCH_HEAD
fatal: refusing to merge unrelated histories

AbduLrhman@AbduLrhman-Sinan MINGW64 ~/localiztion-styles-app/lec01-intro (main)
$ git rm test-file.txt
error: the following file has changes staged in the index:
  test-file.txt
(use --cached to keep the file, or -f to force removal)

AbduLrhman@AbduLrhman-Sinan MINGW64 ~/localiztion-styles-app/lec01-intro (main)
$ echo "محظوظ مُؤقتة" > temp-file.txt
git add temp-file.txt
git rm --cached temp-file.txt
warning: in the working copy of 'temp-file.txt', LF will be replaced by CRLF the next time Git touches it
rm 'temp-file.txt'

AbduLrhman@AbduLrhman-Sinan MINGW64 ~/localiztion-styles-app/lec01-intro (main)
$ echo "محظوظ غير مهم" > untracked-file.txt
git clean -f
Removing temp-file.txt
Removing untracked-file.txt
```

u origin hotfix- git push .1

الوظيفة:

رفع الفرع **hotfix** إلى GitHub لأول مرة، وربطه بالفرع البعيد.

النتيجة:

- تم رفع الملفات (3 كائنات).
- ظهرت رسالة من GitHub: "أنشئ Pull Request لدمج hotfix".
- تم إعداد تتبع الفرع المحلي **hotfix** للفرع البعيد.

`git pull origin main .2` ▲

الوظيفة:

محاولة سحب تغييرات من الفرع `main` على GitHub.

المشكلة:

```
bash  
refusing to merge unrelated histories :fatal
```

السبب:

Git يرفض الدمج لأن:

- الفرع المحلي والفرع البعيد لا يتشاركان نفس البداية (`initial commit`).)
- كل واحد له تاريخ مختلف.

```
bash  
git pull origin main --allow-unrelated-histories
```

`git rm test-file.txt .3` ▲

الوظيفة:

محاولة حذف الملف `test-file.txt`.

المشكلة:

```
bash  
the following file has changes staged in the index :error
```

السبب:

الملف نتم إضافته إلى git add سابقاً، ولا يمكن حذفه بدون توضيح أنك تقصد ذلك.

```
bash  
f test-file.txt- git rm
```

- لإلغاء تتبعه فقط والإبقاء عليه:

```
bash  
git rm --cached test-file.txt
```

`temp-file.txt < echo .4`

أنشأت ملف اسمه `temp-file.txt` يحتوي على النص "محتويات مؤقتة".

```
git add temp-file.txt .5
```

أضفت الملف إلى staging area.

```
git rm --cached temp-file.txt .6
```

أغليت تتبع الملف دون حذفه من الجهاز.

التحذير:

```
bash  
LF will be replaced by CRLF
```

يعني: نهاية الأسطر سيتم تحويلها لطريقة Windows — هذا مجرد تحذير.

```
untracked-file.txt < echo .7
```

أنشأت ملف جديد غير متتبع.

```
f- git clean .8
```

الوظيفة:

حذف كل الملفات غير المتابعة (untracked) في المستودع.

النتيجة:

تم حذف:

```
temp-file.txt •  
untracked-file.txt •
```

وفي الختام، نسأل الله أن ويرزقنا وإياكم الإخلاص في القول والعمل، ويكتب لنا السداد والتوفيق في الدارين.

اللهم علمنا ما ينفعنا، وانفعنا بما علمتنا، وزدنا علماً وعملاً صالحًا، واجعل هذا العلم حجةً لنا لا علينا.
آمين يا رب العالمين.