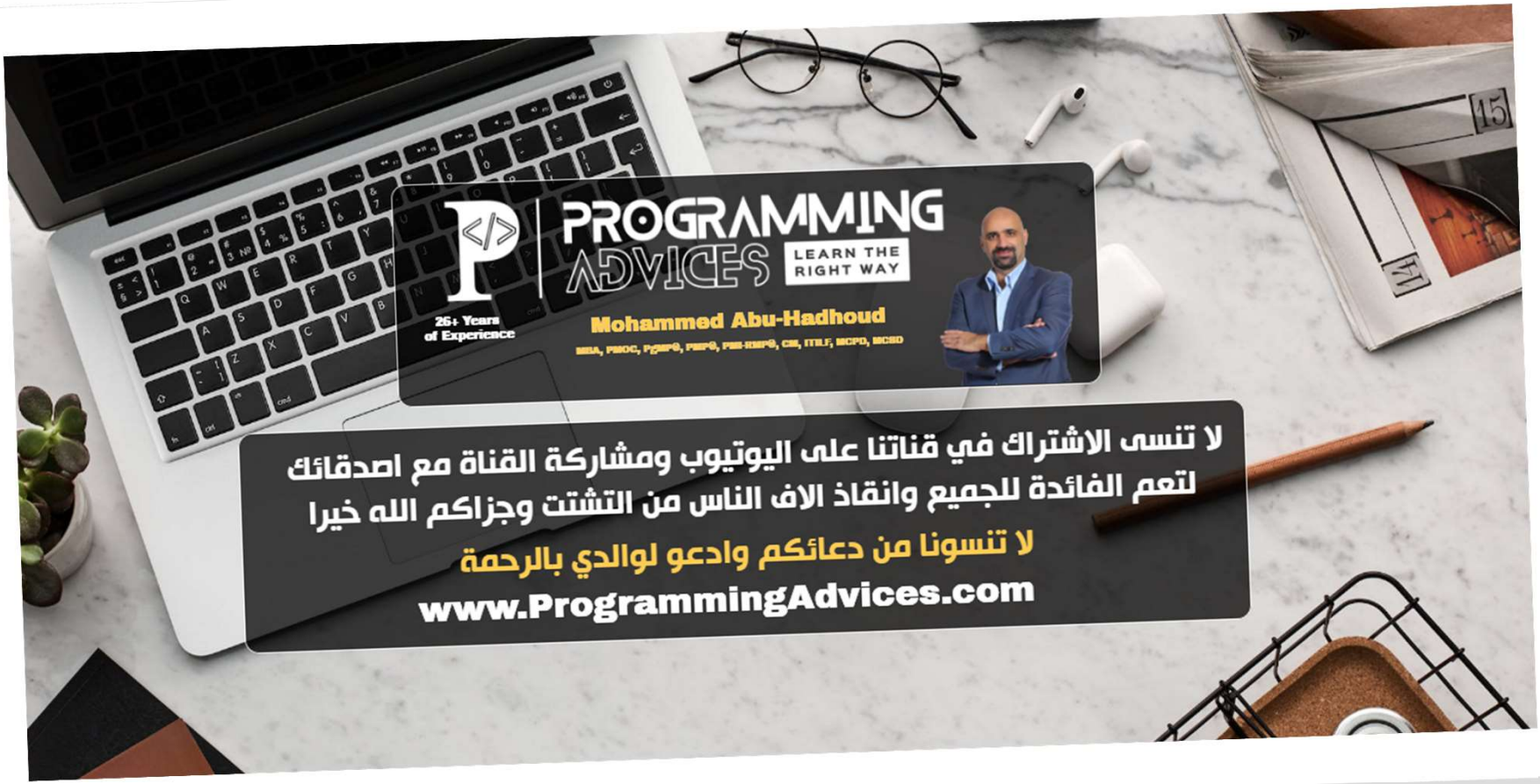
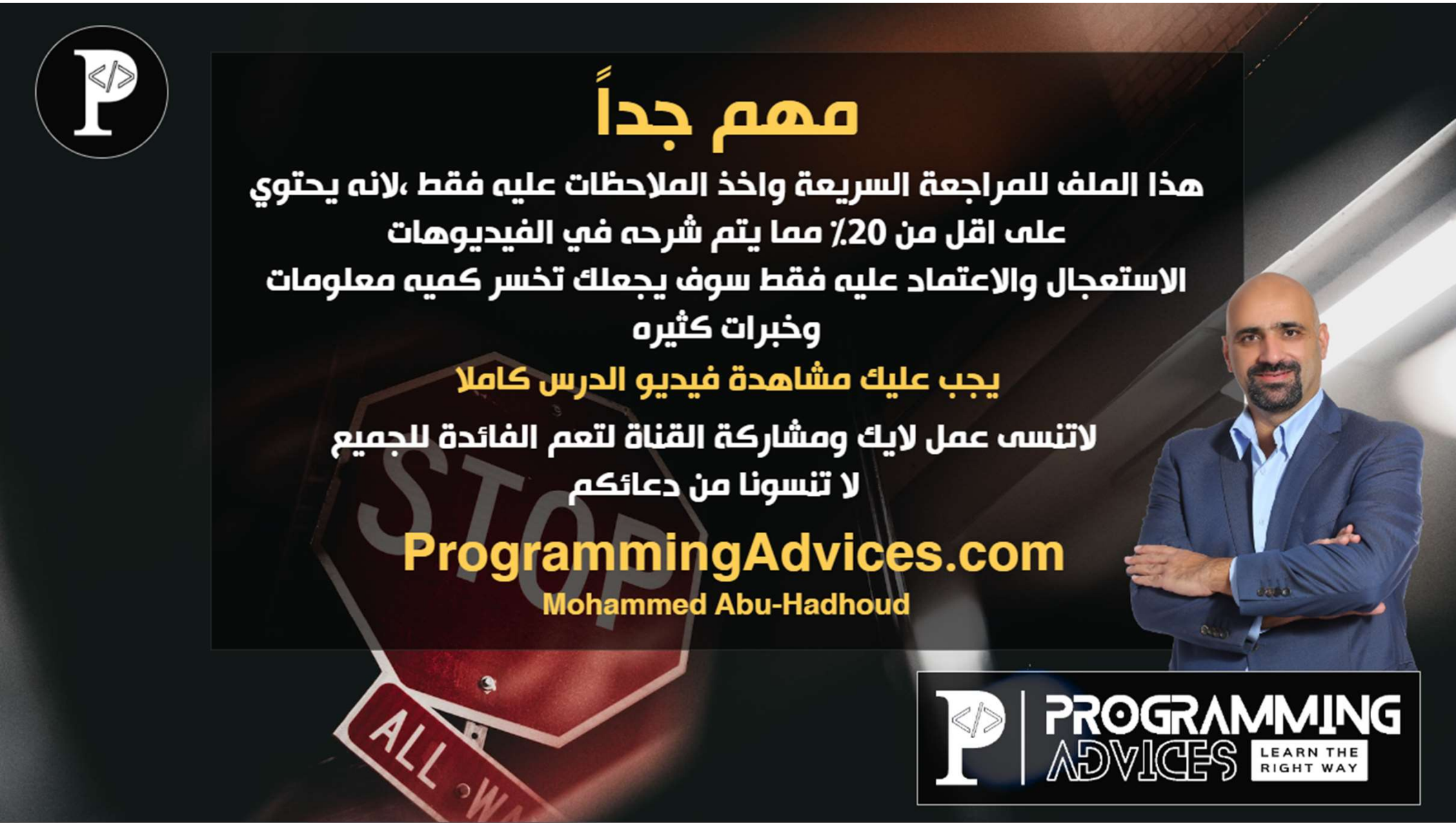
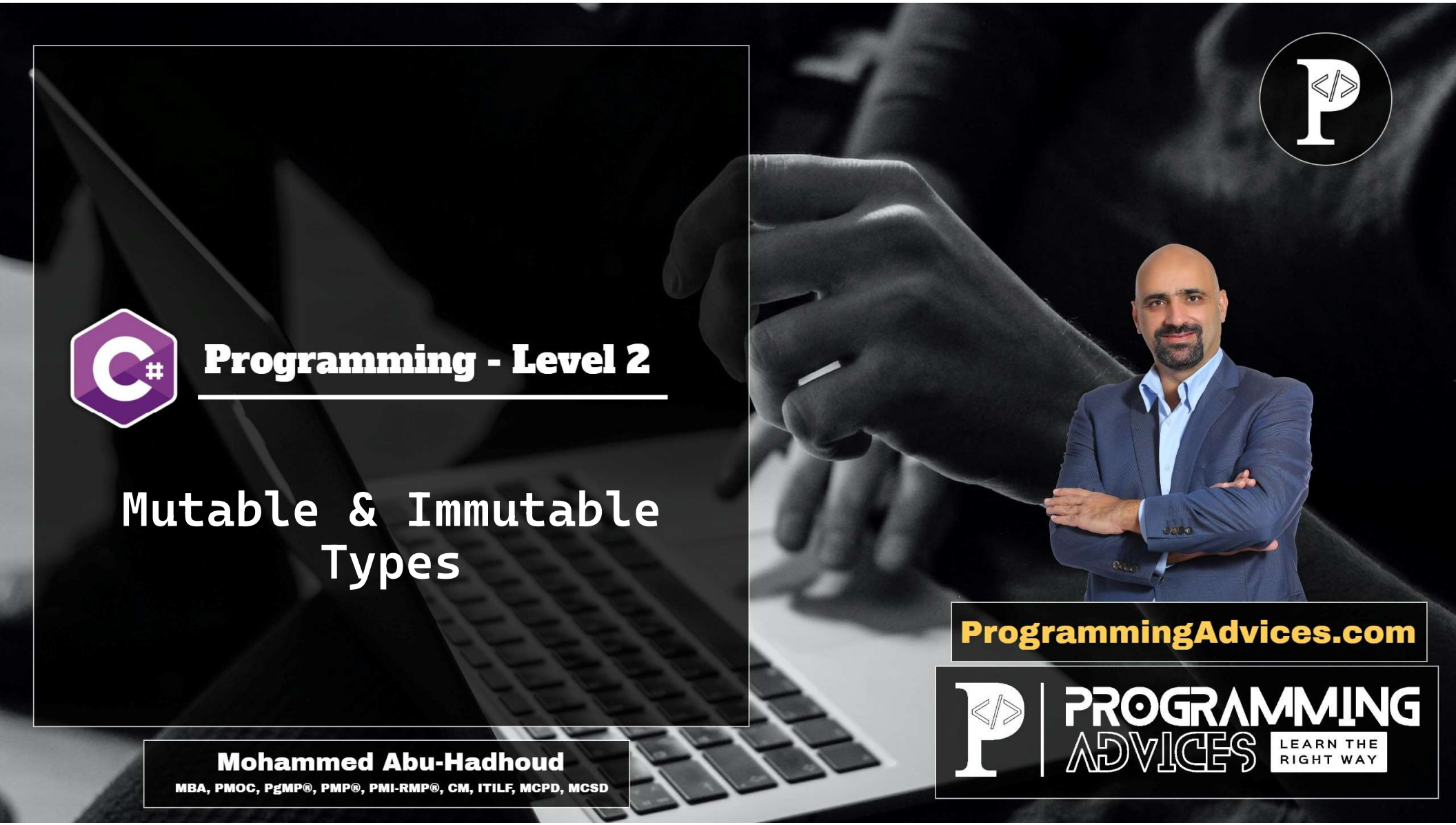

PROGRAMMING ADVICES
LEARN THE RIGHT WAY
26+ Years of Expericace
Mohammed Abu-Hadhoud
MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CSE, ITILF, MCPD, MCSD
لا تنسى الاشتراك في قناتنا على اليوتيوب ومشاركة القناة مع اصدقائك
لتعم الفائدة للجميع وانقاذ الاف الناس من التشتت وجزاكم الله خيرا
لا تنسونا من دعائكم وادعو لوالدي بالرحمة
www.ProgrammingAdvices.com


مهم جداً
هذا الملف للمراجعة السريعة واخذ الملاحظات عليه فقط ،لانه يحتوي
على اقل من 20٪ مما يتم شرحه في الفيديوهات
الاستعجال والاعتماد عليه فقط سوف يجعلك تخسر كميه معلومات
وخبرات كثيره
يجب عليك مشاهدة فيديو الدرس كاملا
لاتنسى عمل لايك ومشاركة القناة لتعم الفائدة للجميع
لا تنسونا من دعائكم
ProgrammingAdvices.com
Mohammed Abu-Hadhoud
STOP
PROGRAMMING
ADVICES
LEARN THE
RIGHT WAY


Programming - Level 2
Mutable & Immutable Types
ProgrammingAdvices.com
PROGRAMMING ADVICES
LEARN THE RIGHT WAY
Mohammed Abu-Hadhoud
MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CM, ITILF, MCPD, MCSD

Mutable and Immutable Types

- In C#, types are classified as either mutable or immutable based on whether their instances can be modified after they are created.

- Understanding the difference between mutable and immutable types is crucial for designing robust and maintainable code.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CM, ITILF, MCPD, MCSD
26+ years of experience

Mutable Types:

- Mutable types are types whose instances can be modified after they are created.
  - Characteristics: Properties or fields of a mutable type can be changed.
  - Changes to an instance affect the state of that instance.
  - Examples include classes, arrays, and custom objects where properties can be modified.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CM, ITILF, MCPD, MCSD
26+ years of experience

Immutable Types:

- Immutable types are types whose instances cannot be modified after they are created.
  - Characteristics: Properties or fields of an immutable type cannot be changed after the instance is created.
  - Any operation that appears to modify the instance actually returns a new instance with the desired changes.
  - Examples include strings, tuples, and some built-in value types.
  - Another Example a Person Class that has everything as ready only.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CM, ITILF, MCPD, MCSD
26+ years of experience

Pros and Cons:

Mutable Types:

- Pros:
  - More flexible for certain scenarios.
  - Can be more memory-efficient if state changes frequently.
- Cons:
  - Prone to unintended side effects.
  - May require additional effort to maintain consistency.

Immutable Types:

- Pros:
  - Safer and less error-prone since instances cannot be modified.
  - Easier to reason about and maintain.
- Cons:
  - Creating a new instance for each modification
  - Can be less memory-efficient for certain scenarios.

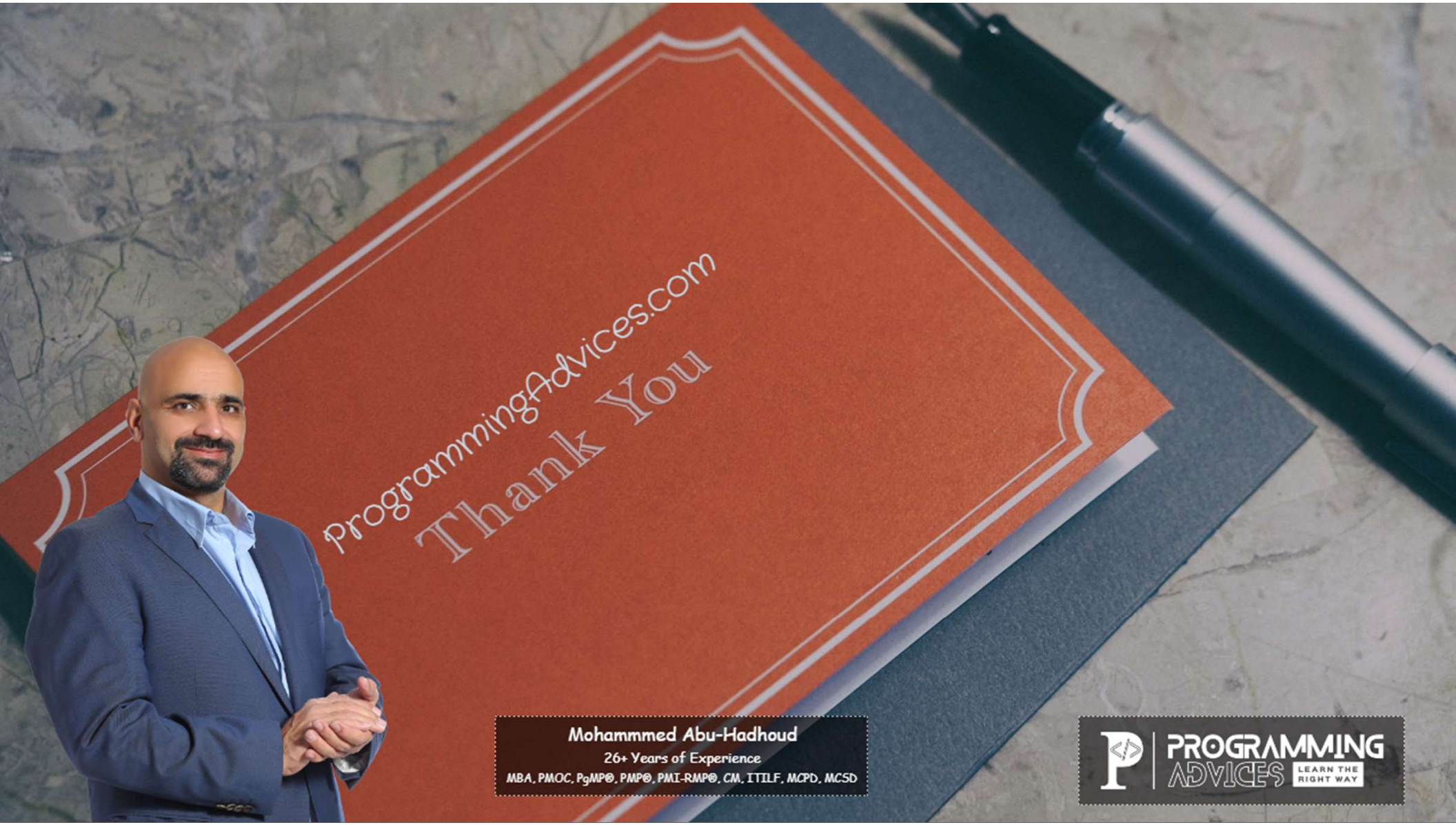
Mohammed Abu-Hadhoud
MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CM, ITILF, MCPD, MCSD
26+ years of experience

Guidelines:

- Favor Immutability: Whenever possible, prefer using immutable types to reduce bugs related to unintended state changes.

- Use Mutability When Necessary: There are scenarios where mutability is more appropriate, such as when frequent state changes are expected or when performance is a critical concern.

Mohammed Abu-Hadhoud
MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CM, ITILF, MCPD, MCSD
26+ years of experience


ProgrammingAdvices.com
Thank You
Mohammmed Abu-Hadhoud
26+ Years of Experience
MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CM, ITILF, MCPD, MCSD
PROGRAMMING ADVICES
LEARN THE RIGHT WAY