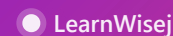


Esercizio di coding con l'IA

Costruisci il progetto GlobalDesk di questo modulo con ChatGPT o Claude partendo da un prompt già pronto che indirizza il modello alla specifica del modulo, poi rivedi, esegui ed estendi quello che ti restituisce.



Lo stesso modulo, costruito come costruirai davvero le cose.

Tutte le altre sezioni di questo modulo spiegano come funziona "Lingue da destra a sinistra e layout a prova di localizzazione". Questa ti chiede di produrlo: dirigendo un assistente con la specifica del modulo stesso, poi verificando che il risultato giri e faccia quello che descrive la traccia del lab. È il flusso di lavoro che userai sul lavoro vero, provato su qualcosa di cui conosci già la risposta giusta.

Che cosa stai chiedendo

Fai sopravvivere GlobalDesk a una cultura che si scrive da destra a sinistra. Aggiungi l'arabo al selettore della lingua e imposta `rightToLeft` su `auto` in `Default.json`, lascia i controlli figli su `RightToLeft.Inherit`, e imposta `RightToLeftLayout` sulla dashboard e sull'editor cliente così che i loro figli vengano rispecchiati. Tieni esattamente un controllo da sinistra a destra, il campo del codice cliente, e spiega in un commento perché quell'identificatore non deve ribaltarsi. Ricostruisci con docking, ancoraggio e un contenitore di layout la parte a posizione fissa del layout, così una traduzione lunga non può troncarsi, aggiungi un file di risorse pseudo-localizzato i cui valori sono allungati e racchiusi tra parentesi, e annota che cosa fa davvero la `DataGridView` dei contatti sotto RTL invece di dare per scontato che venga rispecchiata. Concludi con un elenco di cinque casi di test del layout che puoi rieseguire su qualsiasi schermata.

Il modulo chiede questi deliverable, e il prompt qui sotto li chiede all'assistente esattamente così:

- ✓ Voce araba nel selettore della lingua con `rightToLeft` impostato su `auto` in `Default.json`
- ✓ Dashboard ed editor cliente che usano `RightToLeftLayout` con i figli lasciati su `Inherit`
- ✓ Un controllo deliberatamente da sinistra a destra con un commento che spiega l'eccezione
- ✓ Layout ricostruito con docking, ancoraggio e un contenitore di layout così che le traduzioni lunghe non possano troncarsi
- ✓ Elenco di test RTL che copre testo corto, testo lungo, una finestra stretta, una cultura RTL e il limite della `DataGridView`

Perché è fatto così

GlobalDesk sa già cambiare lingua mentre è in esecuzione: il selettore imposta `Application.CurrentCulture`, il gestore di `Application.CultureChanged` riapplica ogni etichetta che arriva dalle risorse, e il pannello di anteprima riformatta i suoi campioni. Poi un tester apre la dashboard in arabo. Ogni parola è tradotta e ognuna è nel posto sbagliato: le label si stringono al bordo sinistro dei loro campi, la colonna di

navigazione sta dove l'occhio di un lettore arabo arriva per ultimo, e le lunghe etichette tedesche del modulo precedente spingono già un pulsante fuori da un pannello a larghezza fissa.

Questi sono i termini che l'assistente deve azzeccare. Se quello che ti restituisce li usa con leggerezza, non ha capito il modulo:

Termine	Che cosa significa qui
RightToLeft	Una proprietà del controllo con i valori Inherit, No e Yes che dice a un singolo controllo di adattare la direzione del proprio testo e dei propri elementi interni.
RightToLeft.Inherit	La scelta normale per un controllo figlio: segue il genitore, così l'intera schermata cambia direzione a partire da una sola decisione presa in cima.
RightToLeftLayout	Una proprietà diversa che rispecchia la posizione orizzontale dei figli diretti di un contenitore quando RTL è attivo, così la disposizione visiva si ribalta mentre il tuo layout logico resta come l'hai disegnato.
L'impostazione applicativa rightToLeft	Una voce in Default.json; impostata su "auto" fa sì che Wisej.NET deduca il comportamento RTL dalla cultura della sessione invece che da controlli sul nome di cultura sparsi per le tue schermate.
Application.RightToLeft	La stessa decisione espressa nel codice, per le sessioni che devono essere commutate deliberatamente invece che dal rilevamento della cultura.
Espansione del testo	Un'etichetta tradotta ha raramente la lunghezza di quella inglese, ed è per questo che le posizioni fisse in pixel e le larghezze dei pannelli misurate a mano sono le prime a rompersi in una build localizzata.
I limiti RTL documentati	Wisej.NET documenta che DataGridView, ListView e PropertyGrid non supportano pienamente RTL, quindi un piano di test deve nominarli invece di dare per scontato che vengano rispecchiati.

Costruiscilo — scegli il tuo assistente

Entrambi i pulsanti aprono una nuova scheda con lo **stesso prompt già pronto**. Il prompt indirizza il modello alla specifica di questo modulo — un unico file Markdown con la lezione, la traccia del lab e i deliverable — così lavora sul materiale vero invece di tirare a indovinare che cosa sia il Modulo 5. Puoi aprire tu stesso [module-05.md](#) per vedere esattamente che cosa leggerà.



Usa ChatGPT

Apri [chatgpt.com](#) con il prompt di questo modulo già caricato. Funziona con GPT-5 o Codex; qualsiasi piano che possa navigare il web recupererà la specifica.



Usa Claude

Apri [claude.ai](#) con lo stesso prompt già caricato. Chi usa Claude Code può invece incollarlo in un terminale dentro una cartella vuota.



Mandalo così com'è e ottieni un progetto, non un frammento

Il prompt si chiude chiedendo l'intera soluzione in un unico `.zip` con i passi per compilarla ed eseguirla. Scaricalo, apri la soluzione in Visual Studio, premi F5 e GlobalDesk si apre su `localhost`. Ti servono l'SDK .NET e il pacchetto NuGet `Wisej-4`; tutto il resto il prompt chiede all'assistente di scriverlo da sé.

Sto seguendo il Modulo 5 - "Lingue da destra a sinistra e layout a prova di localizzazione" - del corso LearnWisej "Localizzazione in Wisej.NET", e voglio costruire con te il progetto del modulo. Rispondimi in italiano.

Comincia leggendo la specifica del modulo. È un unico file Markdown (in inglese) che contiene l'intera lezione, la traccia del lab e i deliverable:

<https://learnwisej.com/assets/courses/localization-in-wisej-net/specs/module-05.md>

Poi costruisci un progetto Wisej.NET 4 completo che io possa aprire in Visual Studio:

1. Progetto "GlobalDesk", C#, net8.0 o successivo, pacchetto NuGet Wisej-4 4.1.x, host Kestrel che chiama `app.UseWisej()`.
2. Costruisci quello che chiede il lab della specifica: Fai sopravvivere GlobalDesk a una cultura che si scrive da destra a sinistra. Aggiungi l'arabo al selettore della lingua e imposta `rightToLeft` su `auto` in `Default.json`, lascia i controlli figli su `RightToLeft.Inherit`, e imposta `RightToLeftLayout` sulla dashboard e sull'editor cliente così che i loro figli vengano rispecchiati. Tieni esattamente un controllo da sinistra a destra, il campo del codice cliente, e spiega in un commento perché quell'identificatore non deve ribaltarsi. Ricostruisci con docking, ancoraggio e un contenitore di layout la parte a posizione fissa del layout, così una traduzione lunga non può troncarsi, aggiungi un file di risorse pseudo-localizzato i cui valori sono allungati e racchiusi tra parentesi, e annota che cosa fa davvero la `DataGridView` dei contatti sotto RTL invece di dare per scontato che venga rispecchiata. Concludi con un elenco di cinque casi di test del layout che puoi rieseguire su qualsiasi schermata.
3. Produci questi deliverable e dimmi dove si trova ciascuno nella soluzione:
 - Voce araba nel selettore della lingua con `rightToLeft` impostato su `auto` in `Default.json`
 - Dashboard ed editor cliente che usano `RightToLeftLayout` con i figli lasciati su `Inherit`
 - Un controllo deliberatamente da sinistra a destra con un commento che spiega l'eccezione
 - Layout ricostruito con docking, ancoraggio e un contenitore di layout così che le traduzioni lunghe non possano troncarsi
 - Elenco di test RTL che copre testo corto, testo lungo, una finestra stretta, una cultura RTL e il limite della `DataGridView`
4. Segui la specifica ovunque nomini una classe, una proprietà o un evento di Wisej.NET - usa esattamente quello. Dove lascia una scelta aperta, dimmi quale API hai scelto e perché.
5. Tieni l'autorità sul server: stato e validazione vivono in .NET, il browser disegna e riferisce. Non spostare mai una decisione in JavaScript solo perché lì è più comodo.
6. Rendi il risultato dimostrabile: devo poterlo eseguire e vedere il comportamento di cui parla questo modulo, compreso che cosa succede quando qualcosa va storto.

Prima di scrivere codice, riformula con parole tue che cosa costruirai e dove starà ogni pezzo, così posso correggere il progetto finché costa poco.

Quando il codice è pronto, impacchetta l'intera soluzione in un unico file .zip scaricabile e dammi i passi esatti per compilarla ed eseguirla.

Infine rispondi a queste domande della specifica, usando come prova la tua implementazione:

- Un collega imposta `RightToLeft.Yes` su venti controlli e la schermata araba sembra corretta, ma quella tedesca è rimasta invariata. Che cosa ha configurato in realtà, e quale singola proprietà avrebbe fatto il lavoro per l'intero contenitore?
- Il tuo giro pseudo-localizzato mostra tre etichette ancora in inglese puro. Niente si è troncato e niente ha sollevato un'eccezione. Che cosa ti dice questo su quelle tre

etichette, e il lavoro di quale modulo precedente ha una lacuna?

– La griglia dei contatti non viene rispecchiata come il resto della dashboard. Quale delle tue opzioni – accettarlo e documentarlo, sostituire il controllo, oppure scrivere il rispecchiamento a mano – puoi difendere davanti a un revisore, e che cosa dovresti verificare prima di prometterne una qualsiasi?

Chiedimi qualsiasi cosa non ti sia chiara prima di cominciare.

Poi rivedi quello che ti è tornato indietro

Codice che compila non è ancora codice giusto. Le domande interessanti sono domande di progetto: è questo il controllo Wisej.NET che il compito richiede davvero, segue il ciclo di vita che il framework ti offre, lo stato autorevole sta sul server o è stato parcheggiato nel browser perché era comodo? Leggi il risultato da revisore, non da cliente.

Questo modulo ti dà la checklist. Rispondi a ciascuna domanda guardando il codice che hai appena ricevuto:

- ✓ Un collega imposta RightToLeft.Yes su venti controlli e la schermata araba sembra corretta, ma quella tedesca è rimasta invariata. Che cosa ha configurato in realtà, e quale singola proprietà avrebbe fatto il lavoro per l'intero contenitore?
- ✓ Il tuo giro pseudo-localizzato mostra tre etichette ancora in inglese puro. Niente si è troncato e niente ha sollevato un'eccezione. Che cosa ti dice questo su quelle tre etichette, e il lavoro di quale modulo precedente ha una lacuna?
- ✓ La griglia dei contatti non viene rispecchiata come il resto della dashboard. Quale delle tue opzioni — accettarlo e documentarlo, sostituire il controllo, oppure scrivere il rispecchiamento a mano — puoi difendere davanti a un revisore, e che cosa dovresti verificare prima di prometterne una qualsiasi?

Tutto ciò che hai dovuto correggere vale la pena di scriverlo. Una breve nota con le convenzioni e le decisioni a cui sei arrivato, incollata nel primo messaggio della sessione successiva, vale più di qualsiasi prompt ingegnoso.



Per questo corso non c'è ancora un esempio di riferimento — il tuo è il primo

Il [repository degli esempi LearnWisej](#) contiene build già fatte per quasi tutti i corsi, ma non ancora per questo. Significa che non c'è niente con cui confrontarsi e nessuna scorciatoia: la specifica e la tua revisione sono tutta la rete di sicurezza. Tieni quello che costruisci — e se viene bene, ci farebbe piacere vederlo.

Tre prompt per andare oltre

Punta il tuo assistente sul progetto che hai appena costruito e prova questi. Ognuno è un esercizio di revisione ed estensione, non una riscrittura.

- ✓ **Fallo sopravvivere a una brutta giornata.** *"Esamina questo progetto GlobalDesk cercando le modalità di guasto che non gestisce ancora — input sbagliato o mancante, una dipendenza lenta o non disponibile, lavoro che arriva dopo che l'utente ha cambiato pagina, e due azioni che si accavallano. Per ciascuna dimmi prima se è un rischio reale in Wisej.NET 4 e perché, e solo dopo*

scrivi la correzione. Dove decidi che non serve gestire nulla, dillo e spiega il ragionamento invece di aggiungere codice difensivo."

- ✓ **Aggiungi il secondo caso.** *"Estendi questo progetto GlobalDesk con un secondo caso che segue lo stesso schema del primo — un'altra entità, schermata o controllo della stessa forma, collegato allo stesso modo. Estrai in una classe base o in un servizio ciò che è davvero condiviso, ma solo dove se lo merita; due cose simili non chiedono sempre un'astrazione. Mostrami il diff e giustifica ogni cosa che hai estratto."*
- ✓ **Dimostra che funziona davvero.** *"Passa in rassegna questo progetto deliverable per deliverable rispetto alla specifica del modulo su <https://learnwisej.com/assets/courses/localization-in-wisej-net/specs/module-05.md> e mostrami la prova che ciascuno è davvero fatto — il percorso nel codice, su che cosa cliccare e che cosa dovrei vedere. Dove un deliverable è soddisfatto solo in parte, dillo chiaramente invece di arrampicarti sugli specchi. Poi aggiungi quello che manca."*



Portalo dentro il lab

Il lab pratico che chiude questo modulo chiede gli stessi deliverable che hai appena fatto produrre all'assistente. Usa quello che hai qui come forma della tua risposta e come prova a sostegno — non come la risposta in sé. Il lab viene valutato sul tuo ragionamento, e quella parte deve essere tua.