

# CodeHub Frontend TypeScript Interfaces

## Core Data Types

### Language Interface

typescript

```
export interface Language {  
  id: string | number  
  title: string  
  index: number  
  slug: string  
  [key: string]: any  
}
```

### Tutorial Interface

typescript

```
export interface Tutorial {  
  id: string | number  
  title: string  
  slug: string  
  index: number  
  programmingLanguage: string | Language  
  content: string  
  [key: string]: any  
}
```

### Exercise Interface

typescript

```
export interface Exercise {  
  id: string | number  
  title: string  
  slug: string  
  index: number  
  programmingLanguage: string | Language  
  content: string  
  description?: string  
  difficulty?: 'beginner' | 'intermediate' | 'advanced'  
  [key: string]: any  
}
```

## Page Component Props

### Tutorial Pages

typescript

```
export interface TutorialPageProps {  
  params: Promise<{  
    langSlug: string  
    tutSlug: string  
  }>  
}
```

```
export interface LanguagePageProps {  
  params: Promise<{  
    langSlug: string  
  }>  
}
```

```
export interface TutorialPageContentProps {  
  tutorial: Tutorial  
  language: Language  
  allTutorials: Tutorial[]  
  previousTutorial: Tutorial  
  nextTutorial: Tutorial  
}
```

### Exercise Pages

typescript

```
export interface ExercisePageProps {  
  params: Promise<{  
    langSlug: string  
    exSlug: string  
  }>  
}  
  
export interface ExercisePageContentProps {  
  exercise: Exercise  
  language: Language  
  allExercises: Exercise[]  
  previousExercise: Exercise | null  
  nextExercise: Exercise | null  
}
```

## Interactive Pages

typescript

```
export interface InteractivePageProps {  
  params: Promise<{  
    eleSlug: string  
  }>  
}
```

## Component Props

### Tutorial Components

typescript

```
export interface TutorialContentProps {  
  content: string  
}
```

```
export interface TutorialNavigationProps {  
  language: Language  
  previousTutorial: Tutorial | null  
  nextTutorial: Tutorial | null  
}
```

```
export interface TutorialBreadcrumbProps {  
  language: Language  
  tutorial: Tutorial  
}
```

```
export interface TutorialHeaderProps {  
  tutorial: Tutorial  
  language: Language  
  allTutorials: Tutorial[]  
  previousTutorial: Tutorial  
  nextTutorial: Tutorial  
}
```

## Exercise Components

typescript

```
export interface ExerciseContentProps {  
  content: string  
  exercise: Exercise  
}
```

```
export interface ExerciseNavigationProps {  
  language: Language  
  previousExercise: Exercise | null  
  nextExercise: Exercise | null  
}
```

```
export interface ExerciseBreadcrumbProps {  
  language: Language  
  exercise: Exercise  
}
```

```
export interface ExerciseHeaderProps {  
  exercise: Exercise  
  language: Language  
  allExercises: Exercise[]  
  previousExercise: Exercise | null  
  nextExercise: Exercise | null  
}
```

```
export interface ExerciseSidebarProps {  
  language: Language  
  allExercises: Exercise[]  
  currentExercise: Exercise  
}
```

## List Components

typescript

```
export interface TutorialsListProps {  
  tutorials: Tutorial[]  
  language: Language  
  currentTutorial?: Tutorial  
}
```

```
export interface ExercisesListProps {  
  exercises: Exercise[]  
  language: Language  
  currentExercise?: Exercise  
}
```

## Shared Components

typescript

```
export interface LanguageHeaderProps {  
  language: Language  
  itemCount: number  
  itemType: 'tutorials' | 'exercises'  
}
```

```
export interface PageBreadcrumbProps {  
  language: Language  
  item: Tutorial | Exercise  
  itemType: 'tutorial' | 'exercise'  
}
```

## Code Review & Interactive Components

### Code Review

typescript

```
export interface CodeReviewPageProps {  
  language?: string  
  content?: string  
}
```

```
export interface CodeReviewViewProps {  
  language?: string  
  content?: string  
  onClose?: () => void  
  showCloseButton?: boolean  
  className?: string  
}
```

```
export interface Message {  
  sender: string  
  text: string  
}
```

## Layout Components

### Layout Props

typescript

```
export interface LayoutProps {  
  children: React.ReactNode  
  language?: Language  
  showSidebar?: boolean  
}
```

```
export interface SidebarProps {  
  language?: Language  
  currentItem?: Tutorial | Exercise  
  itemType?: 'tutorial' | 'exercise'  
}
```

## Utility Types

### Navigation Types

typescript

```
export interface NavigationItem {  
  title: string  
  slug: string  
  index: number  
  type: 'tutorial' | 'exercise'  
}
```

```
export interface BreadcrumbItem {  
  title: string  
  href?: string  
  isActive?: boolean  
}
```

## Common Props

typescript

```
export interface BaseComponentProps {  
  className?: string  
  children?: React.ReactNode  
}
```

```
export interface WithLanguage {  
  language: Language  
}
```

```
export interface WithNavigation {  
  previousItem?: Tutorial | Exercise | null  
  nextItem?: Tutorial | Exercise | null  
}
```

## File Structure Mapping

Based on your file tree, here's how these interfaces map to your components:

### Learn/[langSlug]/components/

- TutorialBreadcrumb.tsx → TutorialBreadcrumbProps
- TutorialContent.tsx → TutorialContentProps
- TutorialHeader.tsx → TutorialHeaderProps
- TutorialNavigation.tsx → TutorialNavigationProps
- TutorialPageContent.tsx → TutorialPageContentProps



- TutorialSidebar.tsx → TutorialListProps
- LanguageHeader.tsx → LanguageHeaderProps
- TutorialList.tsx → TutorialListProps

## Learn/[langSlug]/[tutSlug]/

- page.tsx → TutorialPageProps

## Learn/[langSlug]/

- page.tsx → LanguagePageProps

## Exercise/[langSlug]/components/

- Similar structure to tutorials but with Exercise-specific props

## Interactives/

- page.tsx → InteractivePageProps

## Usage Examples

### Tutorial Page Implementation

typescript

```
// Learn/[langSlug]/[tutSlug]/page.tsx
export default async function TutorialPage({ params }: TutorialPageProps) {
  const { langSlug, tutSlug } = await params;
  // Implementation
}
```

### Exercise Page Implementation

typescript

```
// Exercise/[langSlug]/[exSlug]/page.tsx
export default async function ExercisePage({ params }: ExercisePageProps) {
  const { langSlug, exSlug } = await params;
  // Implementation
}
```

## Component Usage

typescript

```
// Component implementation
export default function TutorialHeader({
  tutorial,
  language,
  allTutorials,
  previousTutorial,
  nextTutorial
}: TutorialHeaderProps) {
  // Implementation
}
```

## Notes

- All interfaces include `[key: string]: any` for extensibility where appropriate
- Exercise interfaces mirror Tutorial interfaces but are adapted for exercise-specific needs
- Navigation props include null checks for boundary cases (first/last items)
- Props are designed to work with Next.js 13+ App Router patterns
- File structure follows the component organization shown in your tree