



15-3 实现 Provider，掌握其底层实现

render 阶段

beiginWork

```
function updateContextProvider(current: Fiber | null, workInProgress: TypeScript  
  const context = workInProgress.type._context;  
  const newValue = workInProgress.pendingProps.value;  
  
  // 写 context value  
  // pushProvider(context, newValue);  
  
  reconcileChildren(  
    current,  
    workInProgress,  
    workInProgress.pendingProps.children  
  );  
  
  return workInProgress.child;  
}
```

completeWork

✓ 5 changed files	↑ -----	@@ -2,6 +2,7 @@ import { isNum, isStr } from "shared/utils";
✓ examples/src/main.tsx	2 2	import { Fiber } from "../ReactInternalTypes";
✓ packages/react-reconciler/src/ReactFiber.ts	3 3	import {
✓ packages/react-reconciler/src/ReactFiberBeginWork.ts	4 4	ClassComponent,
✓ packages/react-reconcile.../ReactFiberCompleteWork.ts	5 5 +	ContextProvider,
✓ packages/react/src/ReactContext.ts	6 6	Fragment,
	7 7	FunctionComponent,
	8 8	HostComponent,
	↑ -----	@@ -18,6 +19,7 @@ export function completeWork(
	18 19	case Fragment:
	19 20	case ClassComponent:
	20 21	case FunctionComponent:
	22 22 +	case ContextProvider:
	21 23	case HostRoot: {
	22 24	return null;
	23 25	}