

How does RN really work?



```
33
34   ... fetch('https://gabo.re/')
35   ... .then((response) => response.text())
36   ... .then((responseText) => {
37   ...   console.log(responseText)
38   ... })
39   ... .catch((error) => {
40   ...   console.warn(error)
41   ... })
42
```

<ul style="list-style-type: none"> ▼ Development Pods <ul style="list-style-type: none"> ▼ React <ul style="list-style-type: none"> ► Core ► cxxreact ► jschelpers ▼ RCTNetwork <ul style="list-style-type: none"> ▼ Libraries <ul style="list-style-type: none"> ▼ Network <ul style="list-style-type: none"> RCTDataR...stHandler.h RCTDataR...tHandler.m RCTFileRe...stHandler.h RCTFileRe...tHandler.m RCTHTTP...stHandler.h RCTHTTP...andler.mm RCTNetInfo.h RCTNetInfo.m RCTNetworking.h RCTNetworking.mm RCTNetworkTask.h RCTNetworkTask.m ► RCTText ► RCTWebSocket ► Support Files ► yoga 	<pre> 513 task.responseBlock = responseBlock; 514 task.uploadProgressBlock = uploadProgressBlock; 515 516 if (task.requestID) { 517 if (!_tasksByRequestID) { 518 _tasksByRequestID = [NSMutableDictionary new]; 519 } 520 _tasksByRequestID[task.requestID] = task; 521 responseSender(@[task.requestID]); 522 } 523 524 [task start]; 525 } 526 527 #pragma mark - Public API 528 529 - (RCTNetworkTask *)networkTaskWithRequest:(NSURLRequest *)request completionBlock: 530 (RCTURLRequestCompletionBlock)completionBlock; 531 { 532 id<RCTURLRequestHandler> handler = [self handlerForRequest:request]; 533 if (!handler) { 534 RCTLogError(@"No suitable URL request handler found for %@", request.URL); 535 return nil; 536 } 537 RCTNetworkTask *task = [[RCTNetworkTask alloc] initWithRequest:request 538 handler:handler 539 callbackQueue:_methodQueue]; 540 task.completionBlock = completionBlock; 541 return task; 542 } 543 544 #pragma mark - JS API 545 546 RCT_EXPORT_METHOD(sendRequest:(NSDictionary *)query 547 responseSender:(RCTResponseSenderBlock)responseSender) 548 { </pre>
--	--