

Bamboo 빌드 플랜 구성 - Task 설정

이 문서는 Bamboo 빌드 플랜 구성 단계 중에 Task 설정에 대한 정보를 공유하기 위해 작성되었다.

- 빌드 플랜 수정
- Task - Source Code Checkout
- Task - Ant
- 빌드 Artifacts 만들기
- Plan 실행
- 빌드 결과

빌드 플랜 수정

빌드 플랜에서 테스트를 추가하기 위해 아래와 같이 Stages > Default Job을 선택한다.

The screenshot shows the Bamboo web interface for 'Plan configuration' under 'Test Project / Development Build'. The 'Stages' tab is selected, and the 'Default Job' is highlighted in the left sidebar. The main content area shows a list of jobs: 'Default Stage' and 'Default Job'. Below the jobs list, there is a section for 'Related deployment projects' with a 'Create deployment project' button.

Task - Source Code Checkout

Task 설정 페이지가 나타나면, Add task 버튼을 클릭한다. Task types 팝업 창이 나타나면, code를 검색해서 Source Code Checkout 을 선택한다.

The screenshot shows the Bamboo web interface for configuring a plan. The main navigation bar includes 'Bamboo', 'My Bamboo', 'Projects', 'Build', 'Deploy', 'Specs', 'Reports', and 'Create'. The current page is 'Plan configuration' for a 'Development Build' in the 'Default Job' stage. A 'Task types' dialog box is open, displaying a list of tasks. The 'Source Code Checkout' task is highlighted with a red box. The dialog also includes a search bar with the text 'code' and a 'Cancel' button at the bottom right. The background shows the 'Default Job' configuration page with various sections like 'Builder', 'Tests', 'Deployment', 'Source Control', and 'Variables'.

아래와 같이 저장소를 선택하고 하단 Save 버튼을 클릭한다.

Default plan configuration

Stages & jobs 1

Default Stage

Default Job

Job details Docker **Tasks** Requirements Artifacts Other

Tasks

A task is a piece of work that is being executed as part of the build. The execution of a script, a shell command, an Ant Task or a Maven goal are only few examples of Tasks. [Learn more about tasks.](#)

You can use [runtime](#), [plan](#), [project](#) and [global variables](#) to parameterize your tasks.

1 agent has the capabilities to run this job

Source Code Checkout

Final tasks Are always executed even if a previous task fails

Drag tasks here to make them final

Add task

Source Code Checkout configuration

[How to use the Source Code Checkout task](#)

Task description

Disable this task

Add condition to task [?](#)

You can check out one or more repositories with this Task. You can choose to check out the Plan's *Default Repository* or specify a *Specific Repository*. You can add additional repositories to this Plan via the [Plan configuration](#).

Repository*

Default always points to Plans default repository.

Checkout Directory

(Optional) Specify an alternative sub-directory to which the code will be checked out.

Force Clean Build

Removes the source directory and checks it out again prior to each build. This may significantly increase build times.

[+ Add repository](#)

[Save](#) [Cancel](#)

Task - Ant

이 번에는 Ant 빌드를 위해 다음과 같이 정보를 입력 후에 저장한다.

- Executable - ant 선택
- Build file - build.xml
- Target - prepare, compile, test
- The build will produce test results. 체크
- Specify custom results directories : 테스트 보고서 위치 `**/testReports/*.xml`

Source Code Checkout
Checkout Default Repository

Ant

Final tasks Are always executed even if a previous task fails

Drag tasks here to make them final

Add task

Ant configuration How to use the Ant task

Task description

Disable this task

Executable

apache-ant-1.10.5 Add new executable

Build file

build.xml

Target*

prepare compile test

The target you want to execute. You can also define system properties such as -Djava.Awt.Headless=true.

Build JDK*

JDK 1.8.0_191 Add new JDK

Which JDK do you need to use for the build? the JAVA_HOME will be added as an environment variable.

Environment variables

Extra environment variables. e.g. JAVA_OPTS="-Xmx256m -Xms128m". You can add multiple parameters separated by a space.

Working subdirectory

Specify an alternative subdirectory as working directory for the task.

Where should Bamboo look for the test result files?

The build will produce test results.

If checked, the build will fail if no tests are found. Test output must be in JUnit XML format.

Specify custom results directories

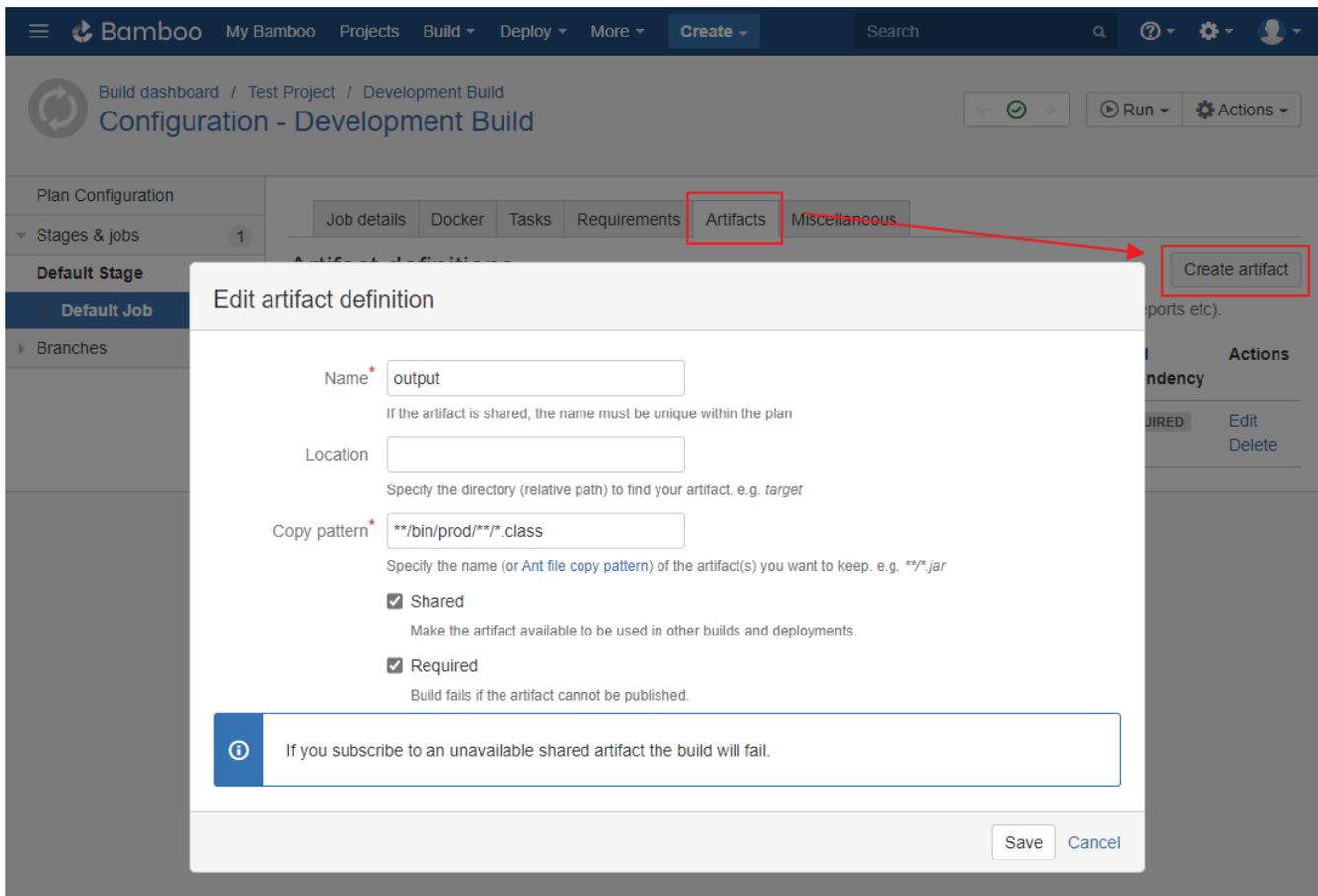
Where does the build place generated test results?
this is a comma separated list of test result directories. You can also use Ant style patterns such as **/test-reports/*.xml

빌드 Artifacts 만들기

빌드 후에 생성되는 파일들을 Artifact로 지정하여, 배포 시 사용할 수 있다. 이를 위해 상단 Artifacts 탭에서 Create artifact 버튼을 클릭한다.

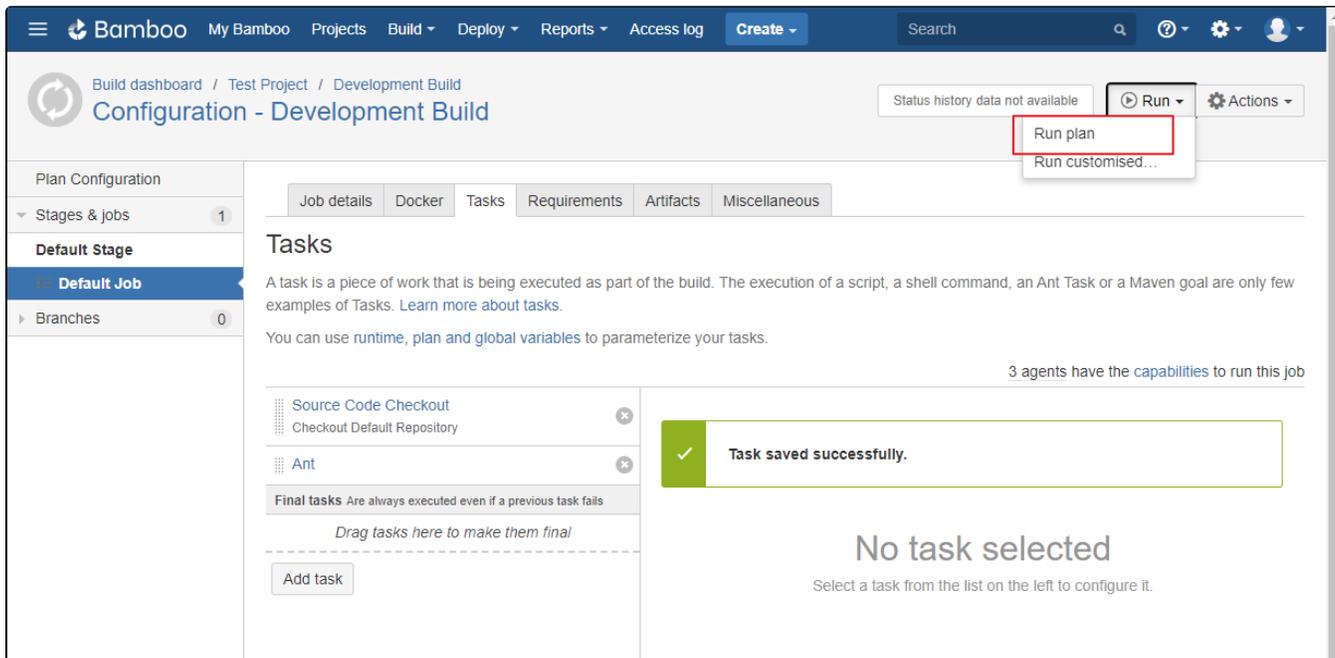
다음 정보를 입력하여 Artifact를 정의 후에 저장한다.

- Name : Artifact의 이름
- Location : Artifact가 만들어지는 경로
- Copy Pattern : 예) **/bin/prod/**/*.*.class
- Shared 체크
- Required 체크



Plan 실행

Task 저장 후에 플랜을 실행하기 위해 상단 Run 메뉴에서 Run plan 메뉴를 선택한다.



빌드 결과

빌드가 성공되면 아래와 같이 빌드 정보를 확인할 수 있다.

The screenshot displays the Bamboo build dashboard for a successful build. The top navigation bar includes 'Bamboo', 'My Bamboo', 'Projects', 'Build', 'Deploy', 'Reports', 'Access log', and 'Create'. The breadcrumb path is 'Build dashboard / Test Project / Development Build'. The main header shows 'Build #1' with a green status indicator and 'Run' and 'Actions' buttons. A green banner at the top states '#1 was successful - Manual run by 설전호 이사'. Below this, there are tabs for 'Summary', 'Tests', 'Commits', 'Artifacts', 'Logs', 'Metadata', and 'Issues'. The 'Summary' tab is active, showing a 'Build result summary' section with 'Details' and 'Tests' subsections. The 'Details' section lists: Completed 17 Mar 2022, 7:54:30 PM - 41 seconds ago; Duration 1 second; Labels None; Agent Second Local Agent; Revision 14c21531...; and Total tests 4. The 'Tests' section shows three metrics: New failures (0), Existing failures (0), and Fixed (0). At the bottom, there is a 'Write a comment...' input field.

- Summary - 빌드 요약 정보
- Tests - 단위 테스트 결과
- Commits - 커밋 정보
- Artifacts - 빌드 산출물
- Logs - 로그 파일