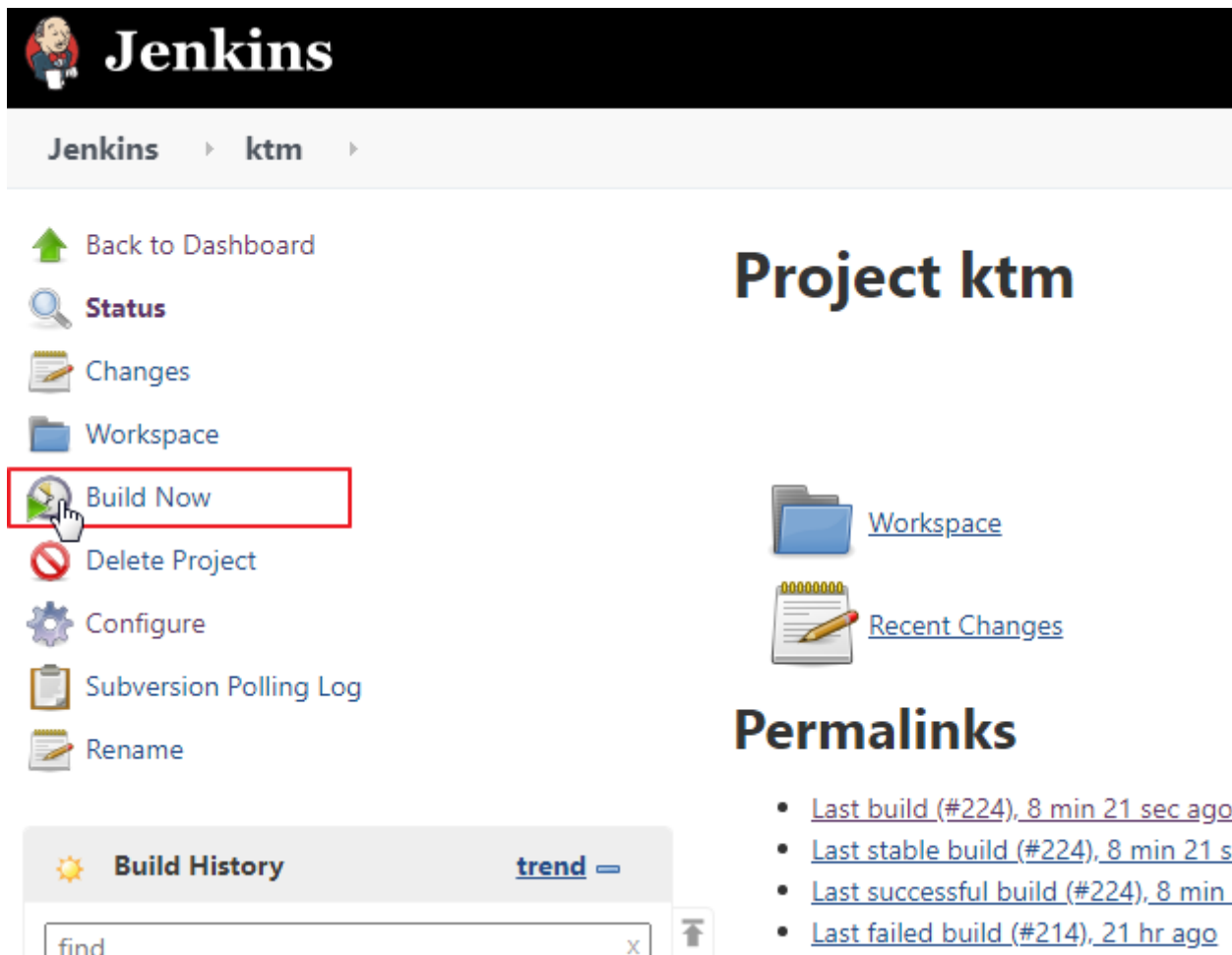


웹소스 운영서버 반영방법

1. SVN 커밋 마감 후 젠킨스 Build 실행

개발서버 소스 기준으로 운영에 반영하기 때문에 젠킨스 빌드를 해주어야함

1. 젠킨스 Build Now 클릭



Jenkins ▸ **ktm** ▸

Back to Dashboard

Status

Changes

Workspace

Build Now

Delete Project

Configure

Subversion Polling Log

Rename

Project ktm

Workspace

Recent Changes

Permalinks

- [Last build \(#224\), 8 min 21 sec ago](#)
- [Last stable build \(#224\), 8 min 21 s](#)
- [Last successful build \(#224\), 8 min](#)
- [Last failed build \(#214\), 21 hr ago](#)

Build History trend ▾

find x

2. 빌드 완료됨을 확인 (파란색)

 **#224** Nov 25, 2020 5:35 PM

2. Nodejs 실행

1. 개발서버 노트북에서 `D:\KTM_Project\송영진\82.svn 읽어 체크` 경로에 감

1. `D:\KTM_Project\송영진\82.svn 읽어 체크\반영생성 log.txt` 파일에서 최근에 반영한 날짜확인

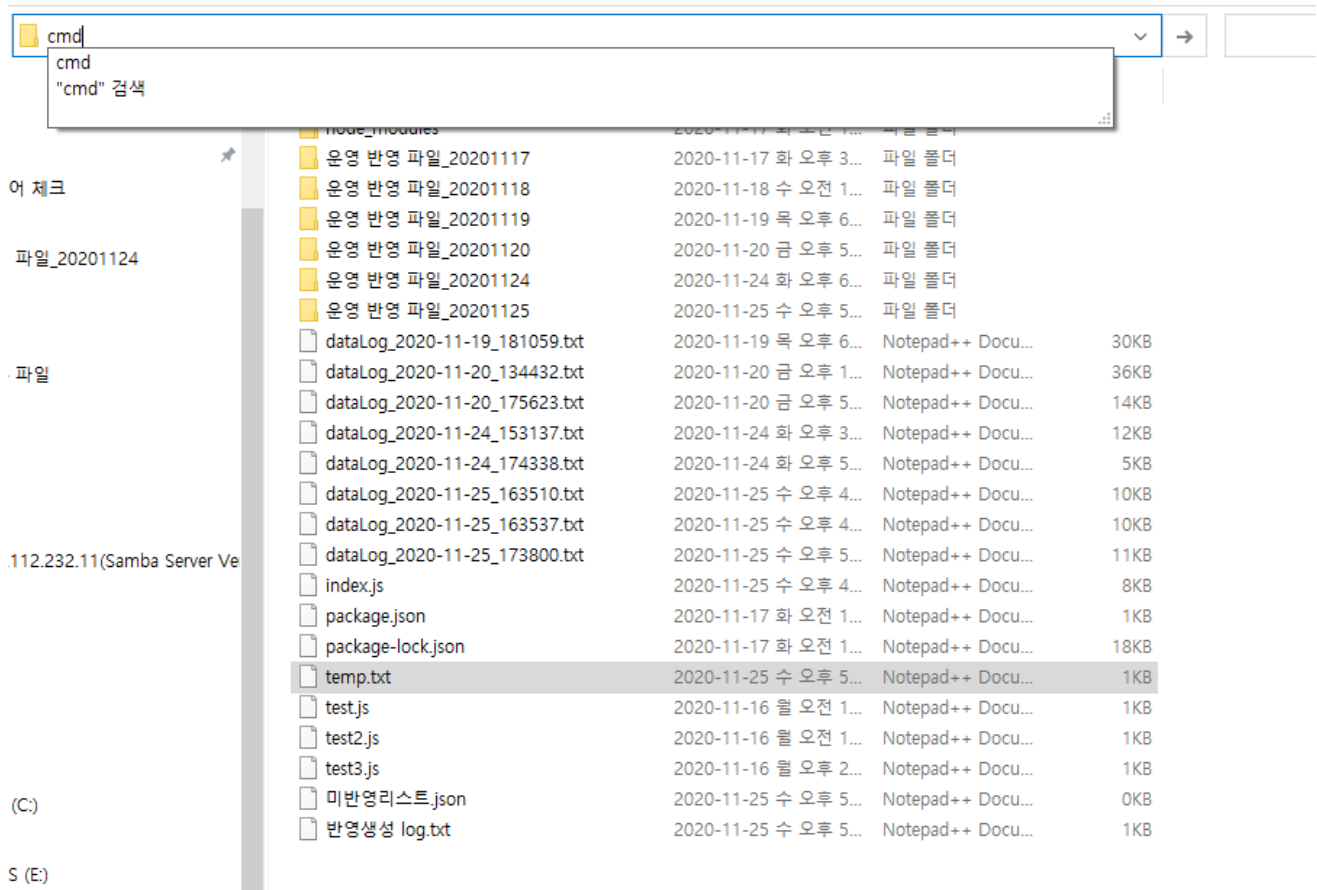
svn에서 (From Date) ~ (현재) 까지 변경된 정보를 가져올 것이기 때문에 기준일자 필요함

```
2020-11-19 17:54 -- 반영됨
2020-11-20 13:45
2020-11-20 17:31 -- 반영됨
2020-11-20 17:57
2020-11-24 14:20
2020-11-24 14:23
2020-11-24 15:28
2020-11-24 15:32 -- 반영됨
2020-11-24 17:44 -- 추가요청
2020-11-25 16:36
2020-11-25 16:36
2020-11-25 17:39 -- 반영됨
2020-11-25 17:51
```

3. `D:\KTM_Project\송영진\82.svn 읽어 체크\index.js` 파일에서 `STD_FROM_DATE` 변수에 반영 기준일자 수정

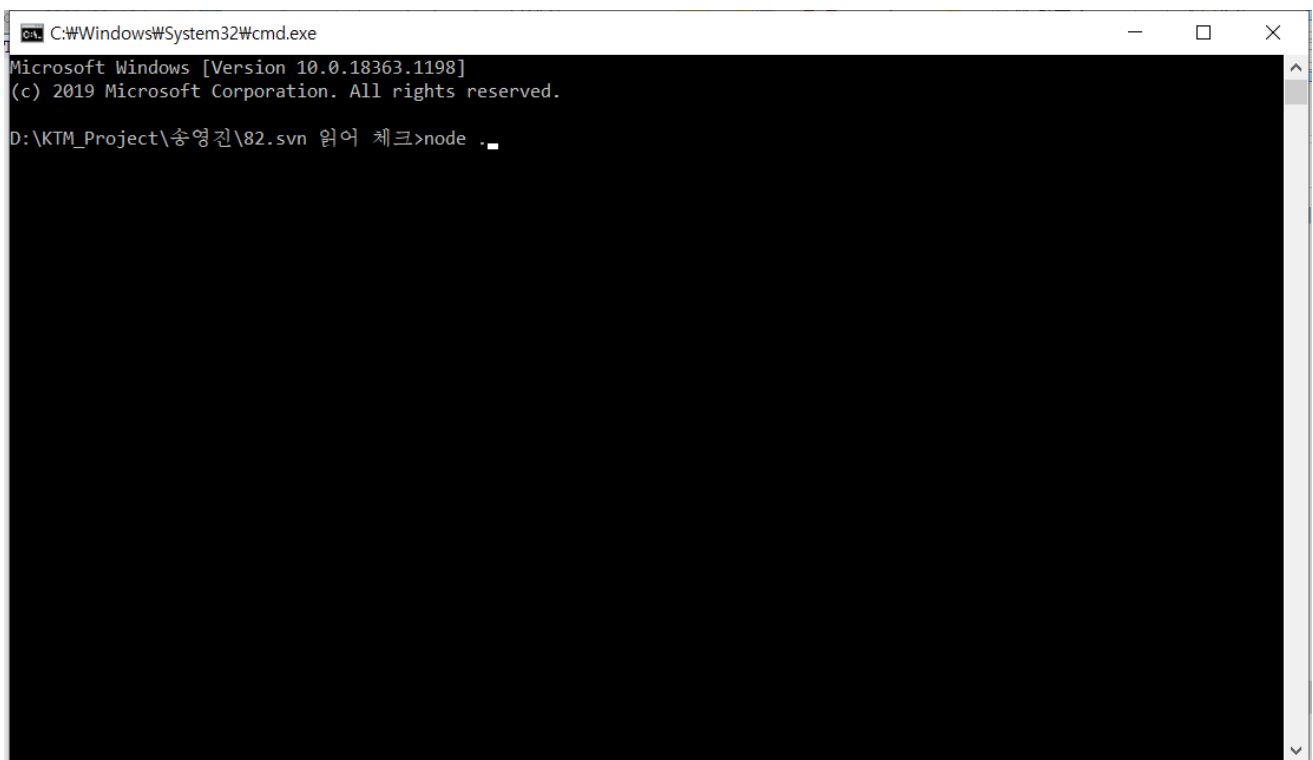
```
1  const cmd = require("node-cmd");
2  const Path = require("path");
3  const fs = require("fs");
4  //require("colors");
5  const fse = require("fs-extra");
6  const archiver = require("archiver");
7  const moment = require("moment");
8  moment.locale("ko");
9
10 const SVN_WORKSPACE =
11   "C:\\WINDOWS\\system32\\config\\systemprofile\\AppData\\Local\\jenkins.jenkins\\workspace\\ktm
12  const STD_FROM_DATE = "2020-11-25 17:39"; // 기준일자 YYYY-MM-DD HH24:MI
13
14  const YYYYMMDD = moment().format("YYYYMMDD"); // YYYYMMDD
15  const DAY_OF_WEEK = moment().format("dddd"); // x요일
16  const STD_TO_DATE = moment().add(1, "minute").format("YYYY-MM-DD HH:mm"); // 1분 후 시간
17
18  const BASE = `운영 반영 파일_${YYYYMMDD}`;
19  const FOLDER_NAME = "웹소스";
20  const SRC_DIR =
21   "C:\\Program Files\\Apache Software Foundation\\Tomcat.9.0\\webapps\\ROOT";
22  const DEST_DIR = Path.join(BASE, FOLDER_NAME);
23
24  const deleteList = []; // 삭제 리스트
25  const exceptList = []; // 제외 리스트
```

4. 파일 주소창에 cmd 입력



5. node . 입력 (뒤에 점 있음)

nodejs 실행



6. 완료 확인

```
C:\Windows\System32\cmd.exe
\WEB-INF\classes\com\hr\eis\empSituation\empTypeGrpSta\EmpTypeGrpSta-mapping-query.xml
\WEB-INF\classes\com\hr\hrm\justice\prePrizeMgr\PrePrizeMgr-mapping-query.xml
\WEB-INF\classes\com\hr\hrm\justice\prePrizeMgr\PrePrizeMgrController.class
\WEB-INF\classes\com\hr\hrm\justice\prePrizeMgr\PrePrizeMgrService.class
\WEB-INF\classes\com\hr\hrm\justice\prePrizeMgr
\WEB-INF\classes\com\hr\common\employee\Employee-mapping-query.xml
\WEB-INF\classes\com\hr\common\employee\EmployeeController.class
\WEB-INF\jsp\hrm\justice\prePrizeMgr\prePrizeMgr.jsp
\WEB-INF\jsp\hrm\justice\prePrizeMgr
\WEB-INF\jsp\hrm\other\empCardPrt2\empCardPrt2.jsp
\WEB-INF\jsp\hrm\other\profileCardPrt\profileCardPrt.jsp
\WEB-INF\jsp\common\popup\employeePopup.jsp
\WEB-INF\jsp\common\include\employeeHeader.jsp
\WEB-INF\jsp\org\organization\orgSchemeIBOrgSrch\orgSchemeIBOrgSrch2.jsp
\WEB-INF\jsp\pap\evaluation\app1st2nd\app1st2nd.jsp
\WEB-INF\jsp\ben\occasion\occAppDet\occAppDet.jsp
\WEB-INF\jsp\sys\security\authority\authority.jsp
\WEB-INF\jsp\eis\compareEmp\compareEmp.jsp
\common\js\common.js
\common\js\employeeHeader.js
\html\report\eis\compareEmpNew.mrd
\html\report\hrm\other\PersonProfile.mrd

ALL : 23 A : 6 M : 17 D : 0

251643 total bytes
archiver has been finalized and the output file descriptor has closed.
처리시간: 460.071ms

D:\KTM Project\송영진\82.svn 읽어 체크>
```

3. 운영반영 데이터 확인하기

1. 운영 반영 파일_오늘날짜로 된 폴더에 파일 저장됨을 확인

크 > 운영 반영 파일_20201125			
이름	수정한 날짜	유형	크기
웹소스	2020-11-25 수 오후 5...	파일 폴더	
KT엠모바일 웹소스_20201125.zip	2020-11-25 수 오후 5...	압축(ZIP) 파일	246KB
log.txt	2020-11-25 수 오후 5...	Notepad++ Docu...	3KB

2. log.txt에서 취합 리스트 확인, 소스 확인하기
3. 오케스트라에 zip파일을 업로드하기

4. 운영 반영 후 로그 생성

반영생성 log.txt 에서 운영에 반영했으면 옆에 주석 남기기

클래스파일 취합 안하는 로직 설명

없음