



Jak prawidłowo logować dane aplikacji

Henryk Nowakowski, Support Engineer

Sebastian Rulik, Support Engineer

May 21, 2019

PUBLIC



Team Hack

Wakacyjny program stażowy dla studentów w naszej firmie!

U nas znajdziesz:



Pracę w zespole Scrumowym pod okiem doświadczonych mentorów.



Luźną, przyjazną atmosferę w miejscu pracy.



Technologie powszechnie wykorzystywane w dzisiejszych rozwiązańach.



Wynagrodzenie za swoją pracę,
staż jest płatny!



Projekty stosowane w rzeczywistym świecie biznesu.

Lipiec - Wrzesień 2019

Biuro SAP w Gliwicach



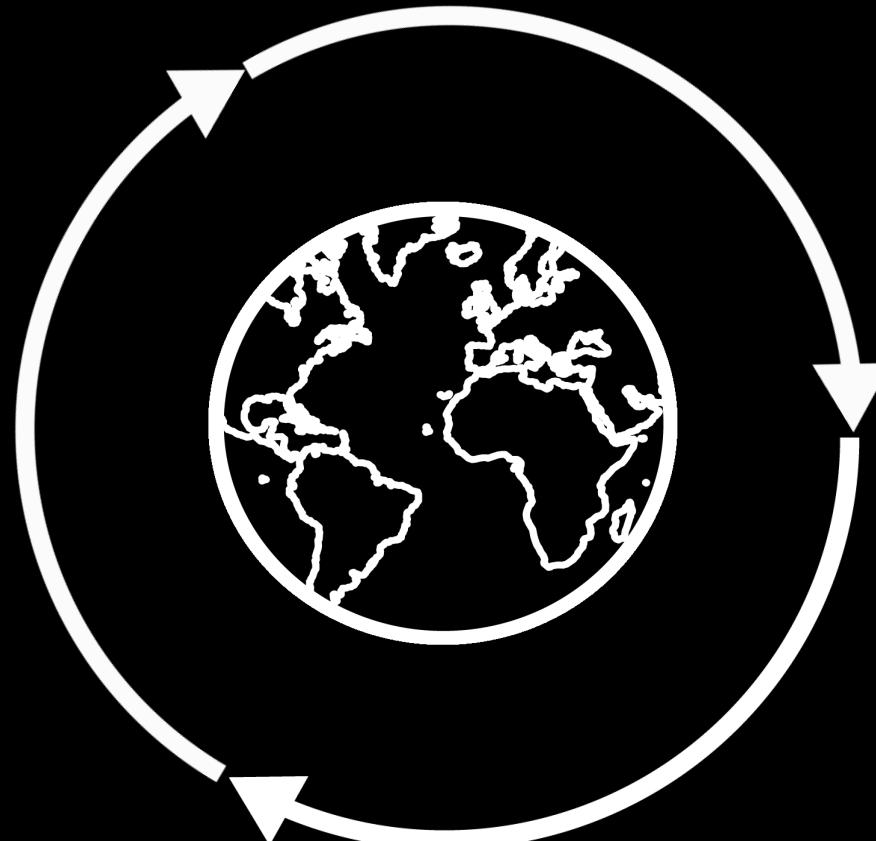
Masz pytanie?
Napisz!
[/Hack.your.career](https://Hack.your.career)

Aplikuj już dziś!
<https://tinyurl.com/y4pvrvyz>



**Top ecommerce,
marketing, billing**

Development: Go, Java,
Cloud Native solutions



> 350 pracowników

**Najlepszy Pracodawca
w rankingu AON**

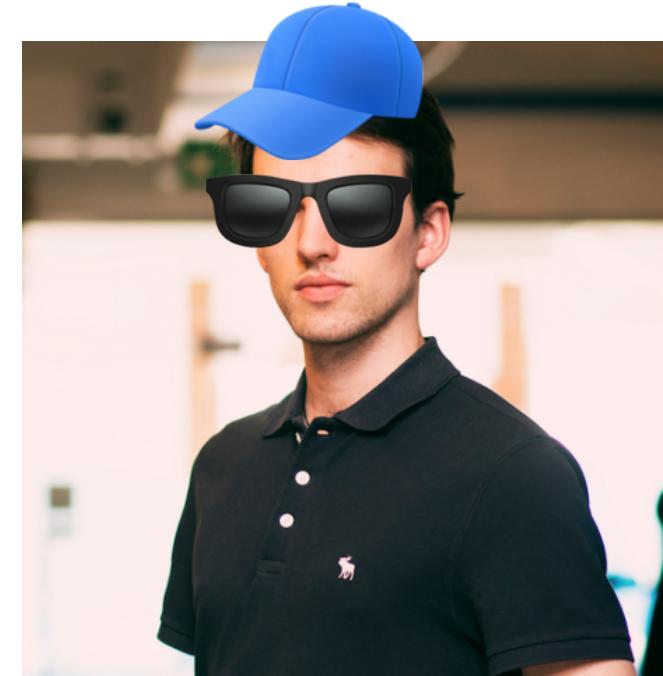
**Jedno z 20 centrów
SAP's Labs Network**

Cloud Services & Support



Sebastian Rulik

Product Support Engineer



Henryk Nowakowski

Product Support Engineer

Agenda

- Wprowadzenie do logowania
- Wyjątki
- Jak korzystać z logów?
- Log4j framework
- Dobre praktyki i ewentualne problemy
- Zarządzanie logami
- ELK (Kibana, Elasticsearch, Logstash)

Szczegółowość logów



Plik z jednego dnia logowania:

Console.log size **354 KB**

VS

Console.log size **3.54 GB**

Logowanie



```
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/customergroups/{groupId}/members],methods=[PUT]" onto public void de.hybris.platform.ycommercewebservices.v2.controller.CustomerGroupsController.setUserListForCustomerGroup(java.lang.String,java.util.List<java.lang.String>)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/customergroups/{groupId}/members],methods=[PUT],consumes=[application/json || application/xml]" onto public void de.hybris.platform.ycommercewebservices.v2.controller.CustomerGroupsController.setUserListForCustomerGroup(java.lang.String,de.hybris.platform.commercewebservicescommons.dto.user.MemberListWsDTO)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/customergroups/{groupId}/members/{userId:+}],methods=[DELETE]" onto public void de.hybris.platform.ycommercewebservices.v2.controller.CustomerGroupsController.removeUsersFromCustomerGroup(java.lang.String,java.lang.String)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/stores],methods=[GET]" onto public de.hybris.platform.commercewebservicescommons.dto.store.StoreFinderSearchPageWsDTO de.hybris.platform.ycommercewebservices.v2.controller.StoresController.locationSearch(java.lang.String,java.lang.Double,java.lang.Double,int,int,java.lang.String,java.lang.String,java.servlet.http.HttpServletResponse) throws de.hybris.platform.commercewebservicescommons.errors.exceptions.RequestParamException
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/stores],methods=[HEAD]" onto public void de.hybris.platform.ycommercewebservices.v2.controller.StoresController.countLocationSearch(java.lang.String,java.lang.Double,java.lang.Double,double,de.hybris.platform.ycommercewebservicescommons.errors.exceptions.RequestParamException)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/stores/{storeId}],methods=[GET]" onto public de.hybris.platform.commercewebservicescommons.dto.store.PointOfServiceWsDTO de.hybris.platform.ycommercewebservices.v2.controller.StoresController.locationDetails(java.lang.String,java.lang.String)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/users/{userId}/carts/{cartId}/save],methods=[PATCH]" onto public de.hybris.platform.commercewebservicescommons.dto.order.SaveCartResultWsDTO de.hybris.platform.ycommercewebservices.v2.controller.SaveCartController.saveCart(java.lang.String,java.lang.String,java.lang.String,java.lang.String) throws de.hybris.platform.commercewebservices.order.CommerceSaveCartException
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/users/{userId}/carts/{cartId}/flagForDeletion],methods=[PATCH]" onto public de.hybris.platform.commercewebservicescommons.dto.order.SaveCartResultWsDTO de.hybris.platform.ycommercewebservices.v2.controller.SaveCartController.flagForDeletion(java.lang.String,java.lang.String) throws de.hybris.platform.commercewebservices.order.CommerceSaveCartException
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/users/{userId}/carts/{cartId}/restoresavedcart],methods=[PATCH]" onto public de.hybris.platform.commercewebservicescommons.dto.order.SaveCartResultWsDTO de.hybris.platform.ycommercewebservices.v2.controller.SaveCartController.restoreSavedCart(java.lang.String,java.lang.String) throws de.hybris.platform.commercewebservices.order.CommerceSaveCartException
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/users/{userId}/carts/{cartId}/savedcart],methods=[GET]" onto public de.hybris.platform.commercewebservicescommons.dto.order.SaveCartResultWsDTO de.hybris.platform.ycommercewebservices.v2.controller.SaveCartController.getSavedCart(java.lang.String,java.lang.String) throws de.hybris.platform.commercewebservices.order.CommerceSaveCartException
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/users/{userId}/carts/{cartId}/clonesavedcart],methods=[POST]" onto public de.hybris.platform.commercewebservicescommons.dto.order.SaveCartResultWsDTO de.hybris.platform.ycommercewebservices.v2.controller.SaveCartController.cloneSaveCart(java.lang.String,java.lang.String,java.lang.String) throws de.hybris.platform.commercewebservices.order.CommerceSaveCartException
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/export/products],methods=[GET]" onto public de.hybris.platform.commercewebservicescommons.dto.product.ProductListWsDTO de.hybris.platform.ycommercewebservices.v2.controller.ExportController.exportProducts(java.lang.String,int,int,java.lang.String,java.lang.String,java.lang.String)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/catalogs/{catalogId}],methods=[GET]" onto public de.hybris.platform.commercewebservicescommons.dto.catalog\CatalogWsDTO de.hybris.platform.ycommercewebservices.v2.controller.CatalogsController.getCatalog(java.lang.String)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/catalogs],methods=[GET]" onto public de.hybris.platform.commercewebservicescommons.dto.catalog\CatalogListWsDTO de.hybris.platform.ycommercewebservices.v2.controller.CatalogsController.getCatalogs(java.lang.String)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/catalogs/{catalogId}/{catalogVersionId}],methods=[GET]" onto public de.hybris.platform.commercewebservicescommons.dto.catalog.CatalogVersionWsDTO de.hybris.platform.ycommercewebservices.v2.controller.CatalogsController.getCatalogVersion(java.lang.String,java.lang.String,java.lang.String)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/catalogs/{catalogId}/categories/{categoryId}],methods=[GET]" onto public de.hybris.platform.commercewebservicescommons.dto.catalog.CategoryHierarchyWsDTO de.hybris.platform.ycommercewebservices.v2.controller.CatalogsController.getCategories(java.lang.String,java.lang.String,int,int,java.lang.String)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/forgottenpasswordtokens],methods=[POST]" onto public void de.hybris.platform.ycommercewebservices.v2.controller.ForgottenPasswordsController.restorePassword(java.lang.String)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{}],methods=[POST]" onto public void de.hybris.platform.ycommercewebservices.v2.controller.DefaultController.defaultRequest(javax.servlet.http.HttpServletRequest)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/deliverycountries],methods=[GET]" onto public de.hybris.platform.commercewebservicescommons.dto.user.CountryListWsDTO de.hybris.platform.ycommercewebservices.v2.controller.MiscsController.getDeliveryCountries(java.lang.String)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/languages],methods=[GET]" onto public de.hybris.platform.commercewebservicescommons.dto.storesession.LanguageListWsDTO de.hybris.platform.ycommercewebservices.v2.controller.MiscsController.getLanguages(java.lang.String)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/currencies],methods=[GET]" onto public de.hybris.platform.commercewebservicescommons.dto.storesession.CurrencyListWsDTO de.hybris.platform.ycommercewebservices.v2.controller.MiscsController.getCurrencies(java.lang.String)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/cardtypes],methods=[GET]" onto public de.hybris.platform.commercewebservicescommons.dto.order.CardTypeListWsDTO de.hybris.platform.ycommercewebservices.v2.controller.MiscsController.getCardTypes(java.lang.String)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/titles],methods=[GET]" onto public de.hybris.platform.commercewebservicescommons.dto.user.TitleListWsDTO de.hybris.platform.ycommercewebservices.v2.controller.MiscsController.getTitles(java.lang.String)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/osauth/confirm_access}],methods=[GET]" onto public org.springframework.web.servlet.ModelAndView de.hybris.platform.oauth2.controller.OAuth2AccessTokenController.getAccessConfirmation(org.springframework.security.oauth2.provider.AuthorizationRequest) throws java.lang.Exception
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/integration/merchant_callback],methods=[POST]" onto public void de.hybris.platform.acceleratorwebservicesaddon.controllers.v2.MerchantCallbackController.handleMerchantCallback(javax.servlet.http.HttpServletRequest)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/integration/users/{userId}/carts/{cartId}/payment/sop/response],methods=[POST]" onto public void de.hybris.platform.acceleratorwebservicesaddon.controllers.v2.MerchantCallbackController.handleMerchantCallback(javax.servlet.http.HttpServletRequest,java.lang.String,java.lang.String)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/users/{userId}/carts/{cartId}/consolidate],methods=[GET]" onto public de.hybris.platform.commercewebservicescommons.dto.store.PointOfServiceListWsDTO de.hybris.platform.acceleratorwebservicesaddon.controllers.v2.ExtendedCartsController.getConsolidatedPickupOptions(java.lang.String)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/users/{userId}/carts/{cartId}/consolidate],methods=[POST]" onto public de.hybris.platform.commercewebservicescommons.dto.order.CartModificationListWsDTO de.hybris.platform.acceleratorwebservicesaddon.controllers.v2.ExtendedCartsController.consolidatePickupLocations(java.lang.String,java.lang.String) throws de.hybris.platform.commercewebservices.order.CommerceCartModificationException
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/users/{userId}/carts/{cartId}/payment/sop/request],methods=[GET]" onto public de.hybris.platform.commercewebservicescommons.dto.payment.PaymentRequestWsDTO de.hybris.platform.acceleratorwebservicesaddon.controllers.v2.ExtendedCartsController.getSopPaymentRequestDetails(java.lang.String,boolean,java.lang.String,java.lang.String,java.lang.String)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/users/{userId}/carts/{cartId}/payment/sop/response],methods=[POST]" onto public de.hybris.platform.commercewebservicescommons.dto.order.PaymentDetailsWsDTO de.hybris.platform.acceleratorwebservicesaddon.controllers.v2.ExtendedCartsController.handleSopPaymentResponse(javax.servlet.http.HttpServletRequest,de.hybris.platform.commercewebservicescommons.dto.payment.SopPaymentDetailsWsDTO,java.lang.String)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/users/{userId}/carts/{cartId}/payment/sop/response],methods=[GET]" onto public de.hybris.platform.commercewebservicescommons.dto.order.PaymentDetailsWsDTO de.hybris.platform.acceleratorwebservicesaddon.controllers.v2.ExtendedCartsController.getSopPaymentResponse(java.lang.String,java.lang.String,javax.servlet.http.HttpServletResponse)
INFO [localhost-startStop-1] [CommerceHandlerMapping] Mapped "[{/baseSiteId}/users/{userId}/carts/{cartId}/payment/sop/response],methods=[DELETE]" onto public void de.hybris.platform.acceleratorwebservicesaddon.controllers.v2.ExtendedCartsController.deleteSopPaymentResponse(java.lang.String)

May 08, 2019 10:02:31 AM org.apache.catalina.startup.TaglibUriRule body
INFO: TLD skipped. URI: http://www.springframework.org/tags is already defined
May 08, 2019 10:02:31 AM org.apache.catalina.startup.TaglibUriRule body
INFO: TLD skipped. URI: http://www.springframework.org/tags/form is already defined
May 08, 2019 10:02:31 AM org.apache.catalina.startup.TldConfig execute
INFO: At least one JAR was scanned for TLDs yet contained no TLDs. Enable debug logging for this logger for a complete list of JARs that were scanned but no TLDs were found in them. Skipping unneeded JARs during scanning can improve startup time and JSP compilation time.
INFO [localhost-startStop-1] [HybrisContextLoaderListener] ****
INFO [localhost-startStop-1] [HybrisContextLoaderListener] adding custom config locations [classpath:productcockpit/productcockpit-spring-configs.xml,classpath:/yacceleratorcockpits/productcockpit/spring/import.xml] for ctx Root WebApplicationContext - /productcockpit: startup date [Thu Jan 01 01:00:00 CET 1970]; parent: hybris Application context - master
INFO [localhost-startStop-1] [HybrisContextLoaderListener] ****
INFO [myRemoveJob]:de.hybris.platform.catalog.jalo.RemoveCatalogVersionJob [myRemoveJob] [MultiThreadedImpExImportReader] Trying to allocate required threads.
INFO [myRemoveJob]:de.hybris.platform.catalog.jalo.RemoveCatalogVersionJob [myRemoveJob] [MultiThreadedImpExImportReader] Required threads have been allocated.
INFO [myRemoveJob]:de.hybris.platform.catalog.jalo.RemoveCatalogVersionJob [myRemoveJob] [MultiThreadedImpExImportReader] Trying to allocate required threads.
INFO [myRemoveJob]:de.hybris.platform.catalog.jalo.RemoveCatalogVersionJob [myRemoveJob] [MultiThreadedImpExImportReader] Required threads have been allocated.
```

May 08, 2019 10:02:31 AM org.apache.catalina.startup.TaglibUriRule body
INFO: TLD skipped. URI: http://www.springframework.org/tags is already defined
May 08, 2019 10:02:31 AM org.apache.catalina.startup.TaglibUriRule body
INFO: TLD skipped. URI: http://www.springframework.org/tags/form is already defined
May 08, 2019 10:02:31 AM org.apache.catalina.startup.TldConfig execute
INFO: At least one JAR was scanned for TLDs yet contained no TLDs. Enable debug logging for this logger for a complete list of JARs that were scanned but no TLDs were found in them. Skipping unneeded JARs during scanning can improve startup time and JSP compilation time.
INFO [localhost-startStop-1] [HybrisContextLoaderListener] ****
INFO [localhost-startStop-1] [HybrisContextLoaderListener] adding custom config locations [classpath:productcockpit/productcockpit-spring-configs.xml,classpath:/yacceleratorcockpits/productcockpit/spring/import.xml] for ctx Root WebApplicationContext - /productcockpit: startup date [Thu Jan 01 01:00:00 CET 1970]; parent: hybris Application context - master
INFO [localhost-startStop-1] [HybrisContextLoaderListener] ****
INFO [myRemoveJob]:de.hybris.platform.catalog.jalo.RemoveCatalogVersionJob [myRemoveJob] [MultiThreadedImpExImportReader] Trying to allocate required threads.
INFO [myRemoveJob]:de.hybris.platform.catalog.jalo.RemoveCatalogVersionJob [myRemoveJob] [MultiThreadedImpExImportReader] Required threads have been allocated.
INFO [myRemoveJob]:de.hybris.platform.catalog.jalo.RemoveCatalogVersionJob [myRemoveJob] [MultiThreadedImpExImportReader] Trying to allocate required threads.
INFO [myRemoveJob]:de.hybris.platform.catalog.jalo.RemoveCatalogVersionJob [myRemoveJob] [MultiThreadedImpExImportReader] Required threads have been allocated.

Log – chronologiczny zapis zawierający informację o zdarzeniach i działaniach dotyczących systemu informatycznego [...]. Log tworzony jest automatycznie przez dany program komputerowy [...].

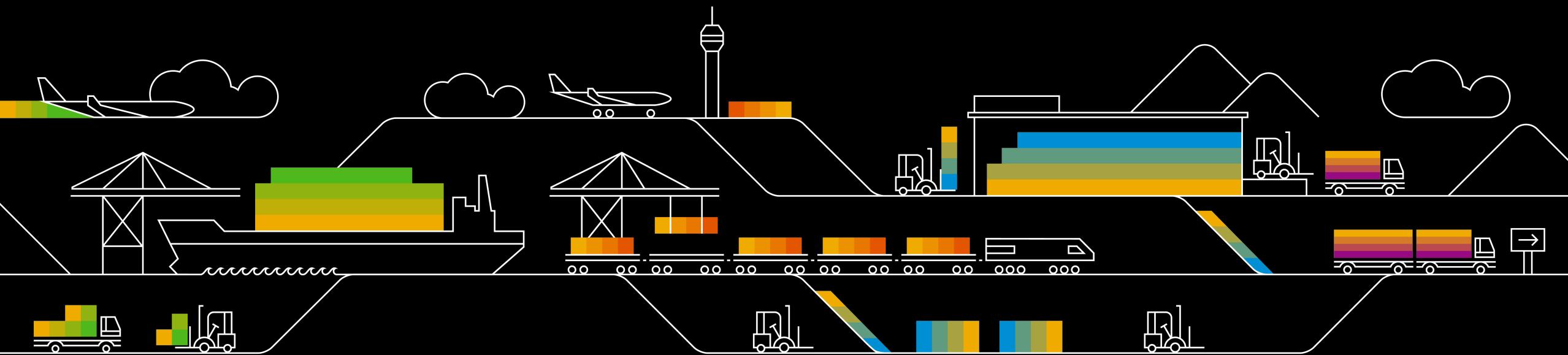
Logi są używane do analizowania pracy systemu informatycznego, np. sporządzania statystyk, wykrywania prób włamań do systemu [...].

Logi mogą być prowadzone mniej lub bardziej szczegółowo [...]

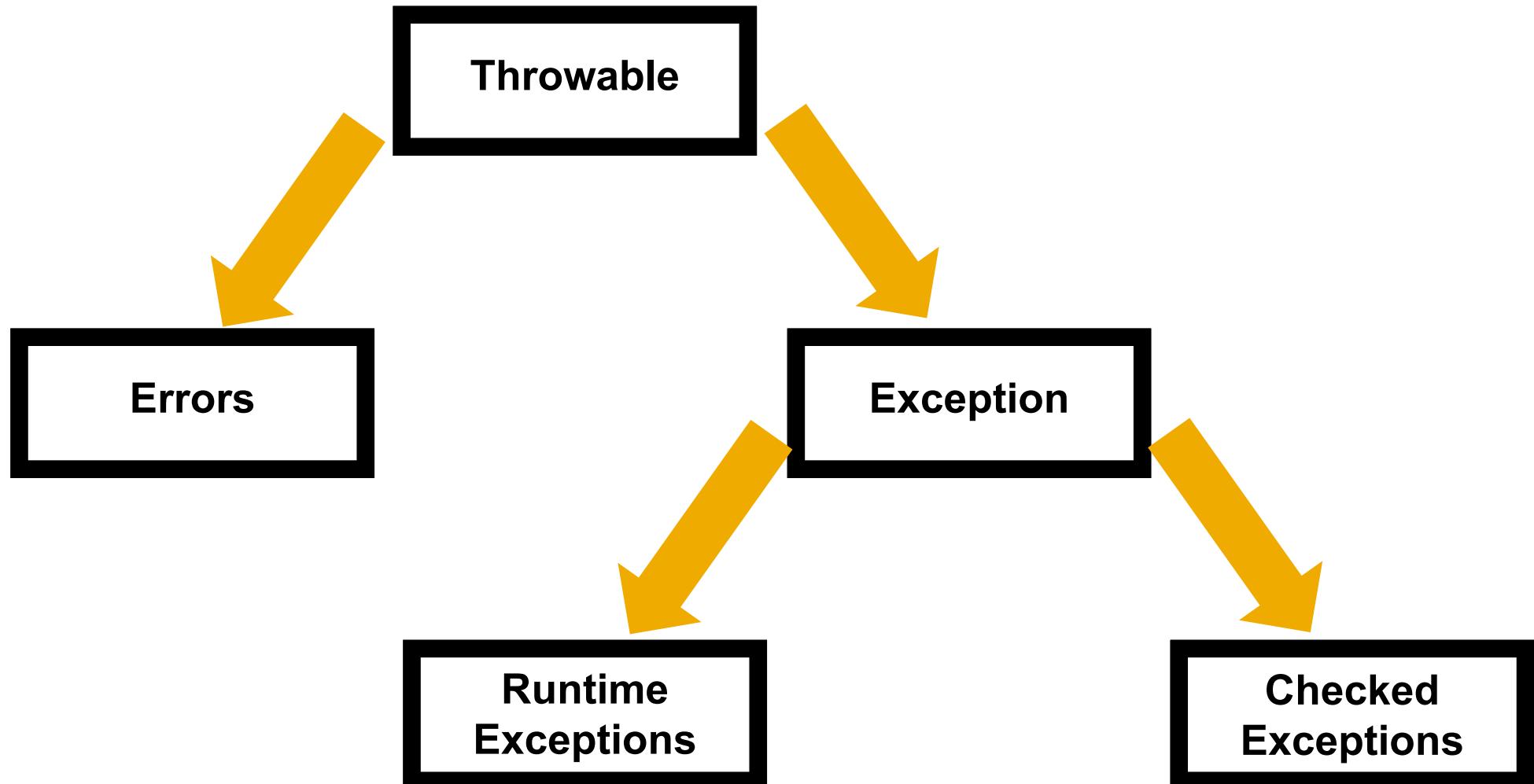
~*Wikipedia*



Logi a wyjątki



O wyjątkach słów kilka



O wyjątkach słów kilka

Wyjątki w javie są obiektami, sygnalizującymi wystąpienie wyjątkowej sytuacji w trakcie działania aplikacji. Zaznaczają, że coś w trakcie działania programu poszło nie tak.

Możemy tworzyć własne i je „**rzucić**” za pomocą klauzuli **throws**.

Do obsługi wyjątków używamy z kolei klauzuli **try / catch**.

```
try {
    // do something
} catch (Exception e) {
    // panic & handle the exception
} finally {
    // close or print stuff
}
```

O wyjątkach słów kilka

```
public class CheckedExceptionsSample {  
    public static void main(String[] args) {  
        CheckedExceptionsSample instance = new CheckedExceptionsSample();  
  
        try {  
            instance.methodWithCheckedException();  
        }  
        catch (IOException e) {  
            e.printStackTrace();  
        }  
    }  
  
    private void methodWithCheckedException() throws IOException {  
        throw new IOException();  
    }  
}
```

Gdzie leży źródło błędów?

Błędna implementacja

- Próba odczytu **nieistniejącego** pliku
- Próba odczytania danych **spoza zakresu tabeli**
- Próba pobrania wartości z **niezainicjalizowanej** zmiennej

Błędne dane podane przez użytkownika

```
try {  
    // do something  
}  
  
catch (Exception e) {  
    // panic  
}  
  
finally {  
    // close or print stuff  
}
```

Najczęściej spotykane błędy

- **NullPointerException** – przy próbie wywołania metody na zmiennej, której wartość wynosi *null*
- **IOException** (wyjątki po nim dziedziczące) – w przypadku problemów z systemem I/O, czyli kiedy wystąpi problem przy pracy z plikami lub z transmisją danych
- **StackOverflowException** – gdy rozmiar stosu przekroczy ilość pamięci zarezerwowanej dla niego (wywoływanie funkcji)
- **OutOfMemoryError** – gdy rozmiar sterty przekroczy ilość pamięci zarezerwowanej dla niego lub skończyła się wolna przestrzeń pamięciowa na dysku (tworzenie obiektów)

Połykanie wyjątków



```
try {  
    //do something  
}  
catch (Exception e) {  
    // crickets  
}
```



```
try {  
    //do something  
}  
catch (Exception e) {  
    log.error("Houston, many problem", e);  
}
```

Dlaczego to takie
ważne?



Szczegółowość w obsłudze wyjątków



Wyjątki JDBC
BatchUpdateException
DataTruncation
SQLException
SQLWarning

Wyjątki dostępu do danych Spring

BadSqlGrammarException

DataAccessException

DuplicateKeyException

CannotAcquireLockException

ConcurrencyFailureException

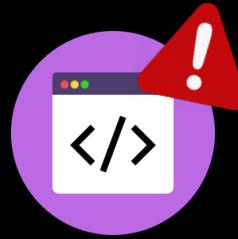
DeadlockLoserDataAccessException

EmptyResultDataAccessException

UncategorizedSQLException

QueryTimeoutException

...i wiele więcej



wyjątki



Słów kilka o logowaniu



Poziomy logowania danych

- **INFO** – logi informacyjne, pokazują progres aplikacji

INFO | [ProductModel] Product with code 12001436 does not exist

INFO | [Licence] Your licence is only for demo usage and is valid for 30 days.

INFO | o.a.s.h.c.SpellCheckComponent: Loading spell index for spellchecker: en

Poziomy logowania danych

- **WARN** – loguje potencjalnie zagrażające aplikacji informacje

WARN | Job will be marked as ABORTED. This could happen if the system is shut down with a hard signal.

WARN | Product is using deprecated java version! Please update java to a newer version

Poziomy logowania danych

- **DEBUG** – szczegółowe logi, które mogą być pomocne w trakcie debugowania aplikacji

DEBUG | Accessing class: [ProductModel]

DEBUG | Trying to connect to: [Application]

DEBUG | (update-shopProducts) Starting update of: summerCatalog

Poziomy logowania danych

- **TRACE** – jeszcze bardziej szczegółowe logowanie, niż DEBUG

TRACE | Saving field UserModel.setUserName in process

TRACE | Saving field UserModel.setUserAddress in process

TRACE | Starting/Stopping servlet produced by bean {beanName}

TRACE | Created XML configuration: conf.xml

Poziomy logowania danych

- **ERROR** – wyznacza błędy, które mogą jednak pozwolić aplikacji wciąż działać

```
ERROR | Transaction rollback: failed to execute query = 'SELECT item.ID FROM product item WHERE (item.productName = ?)
```

```
ERROR | Could not copy data from 88043 to 38942 due to  
org.springframework.dao.DeadlockLoserDataAccessException
```

Poziomy logowania danych

- **FATAL** – bardzo poważne błędy, które mogą prowadzić do przerwania działania aplikacji

FATAL | Out of memory (WRC002). Cannot allocate memory

FATAL | Solr returned an error #500 Permission denied java.io.IOException:
Permission denied

Jak wybrać właściwy
poziom logowania?



Poziomy logowania danych

	FATAL	ERROR	WARN	INFO	DEBUG	TRACE	ALL
OFF							
FATAL	x						
ERROR	x	x					
WARN	x	x	x				
INFO	x	x	x	x			
DEBUG	x	x	x	x	x		
TRACE	x	x	x	x	x	x	
ALL	x	x	x	x	x	x	x

Rodzaje logów



Console Log

```
INFO | jvm 1 | main | 2019/04/06 06:37:09.692 | [101.322.12.104] [ProductModel] product with code  
12001436 does not exist  
INFO | jvm 1 | main | 2019/04/06 06:37:10.793 | [101.322.12.104] [SAPCMSPageDaoImpl] Page with  
qualifier category-C94-en has not been found  
[...]  
  
2019-04-08 07:19:09.100 INFO (qtp1330278544-21) [ ] o.a.s.r.RestManager Registered ManagedResource  
impl org.apache.solr.rest.schema.analysis.ManagedSynonymFilterFactory$SynonymManager for path  
/schema/analysis/synonyms/zh  
2019-04-08 07:19:09.100 INFO (searcherExecutor-38-thread-1-processing-x:master_apparel-de_Product_flip)  
[ x:master_apparel-de_Product_flip] o.a.s.h.c.SpellCheckComponent Loading spell index for spellchecker:  
default  
  
INFO | jvm 1 | main | 2019/04/23 10:56:03.721 |  
[platform.servicelayer.exceptions.ModelRemovalException: could not remove partOf attribute items in  
[alldocuments] for 8799800427107 due to SQL search error : transaction rollback: serialization failure  
query = 'SELECT item_t0.PK FROM medias item_t0 WHERE ( item_t0.p_sourceitem =?)', values =  
[PropertyValue:8799800427107, 8796098822226]  
[...]
```

JDBC Log

```
49|master|100115-14:25:52:837|15 ms|statement|SELECT * FROM attributedescriptors WHERE EnclosingTypePK=?|SELECT * FROM attributedescriptors WHERE EnclosingTypePK=23087628500646720

49|master|100115-14:25:52:837|0 ms|statement|SELECT REALNAME,LANGPK,TYPE1,VALUESTRING1,VALUE1 FROM props WHERE ITEMPK = ? |SELECT REALNAME,LANGPK,TYPE1,VALUESTRING1,VALUE1 FROM props WHERE ITEMPK = 24495003384471968

49|master|100115-14:25:52:837|0 ms|statement|SELECT REALNAME,LANGPK,TYPE1,VALUESTRING1,VALUE1 FROM props WHERE ITEMPK = ? |SELECT REALNAME,LANGPK,TYPE1,VALUESTRING1,VALUE1 FROM props WHERE ITEMPK = 24495003384484224
```

Access Log

```
10.322.12.171 - - [06/Apr/2019:08:36:04 +0200] "GET /api/xCompany/postalCode/search?fields=FULL  
HTTP/1.1" 200 "https://www.xCompany.com/en/p/41003647" "Mozilla/5.0 (Linux; Android 9; SAMSUNG SM-  
G950F Build/PPR1.1808) AppleWebKit/537.36 (KHTML, like Gecko) SamsungBrowser/9.2 Chrome/67.0.3396.87  
Mobile Safari/537.36"  
  
10.322.12.171 - - [06/Apr/2019:08:36:04 +0200] "GET  
/api/xCompany/products/suggestions?max=5&term=glo&fields=FULL HTTP/1.1" 200 26 "-" "Storefront"  
  
10.322.12.171 - - [06/Apr/2019:08:36:04 +0200] "GET /api/xCompany/products/62004419/pickuplocations?  
postalCode=47100&location=Gliwice HTTP/1.1" 404 154411 "-" "SAP Front"  
  
10.322.12.171 - - [06/Apr/2019:08:36:03 +0200] "GET  
/api/xCompany/products/32001644/deliverymodes?fields=FULL HTTP/1.1" 200 1161 "-" "SAP Front"  
  
10.322.12.171 - - [06/Apr/2019:08:36:03 +0200] "GET /api/xCompany/products/32001644/references?  
referenceType=SIMILAR&pageSize=20 HTTP/1.1" 200 25 "-" "SAP Front"
```

Build Log

```
Starting Build...
[INFO] -----
[INFO] Building Maven Stub Project (No POM) 1
[INFO] -----
[INFO] >>> maven-archetype-plugin:2.1:generate (default-cli) @ standalone-pom >> 
[INFO] <<< maven-archetype-plugin:2.1:generate (default-cli) @ standalone-pom << 
[INFO] 
[INFO] Generating project in Batch mode
[INFO] -----
[INFO] Using following parameters for creating project from Old (1.x) Archetype: maven-archetype-quickstart:1.0
[INFO] -----
[INFO] Parameter: groupId, Value: com.mycompany.app
[INFO] Parameter: packageName, Value: com.mycompany.app
[INFO] Parameter: package, Value: com.mycompany.app
[INFO] -----
[INFO] BUILD FAILURE
[INFO] -----
[INFO] Total time: 22.971s
```

Client side logs

The screenshot shows the Chrome DevTools interface with the 'Console' tab selected. The left sidebar displays log levels: 8 messages (No user messages), 4 errors, 4 warnings, 1 info, and 0 verbose. The main area lists the following log entries:

- Failed to load resource: net::ERR_BLOCKED_BY_CLIENT ui:1
- [Deprecation] chrome.loadTimes() is deprecated, instead use standardized API: nextHopProtocol in Navigation Timing 2. <https://www.chromestatus.com/features/5637885046816768>.
- [Deprecation] chrome.loadTimes() is deprecated, instead use standardized API: nextHopProtocol in Navigation Timing 2. <https://www.chromestatus.com/features/5637885046816768>.
- [Deprecation] chrome.loadTimes() is deprecated, instead use standardized API: nextHopProtocol in Navigation Timing 2. <https://www.chromestatus.com/features/5637885046816768>.
- [Deprecation] chrome.loadTimes() is deprecated, instead use standardized API: nextHopProtocol in Navigation Timing 2. <https://www.chromestatus.com/features/5637885046816768>.
- Failed to load resource: net::ERR_BLOCKED_BY_CLIENT dn.js:1
- Uncaught ReferenceError: gbar is not defined at onload ((index):6) (index):6
- Unchecked runtime.lastError: The message port closed before a response was received. (index):1

Inne rodzaje logów...

- solr.log
- audit.log
- wifi.log
- system_gc.log
- Logi w systemach operacyjnych
- Logi w wszelkiego rodzaju narzędziach (Audacity, VLC, MS Office)
- security.log
- database.log
- ...i wiele więcej

A co z nazewnictwem?



- Podział na poziomy logowania
- Podział na pliki
- Podział na rodzaje logów
- Nazewnictwo



wyjątki

Jak tym zarządzać?



node.js



Czym jest Log4J?

C#



Logging

```
public class HelloWorld{  
  
    static LogLevel logLevel = new LogLevel();  
  
    public static void main(String[] args) {  
        System.out.println("Sample console message");  
        //operations...  
  
        if(logLevel.getLevel().equals("DEBUG")) {  
            //extended logging  
            System.out.println("Sample console message for DEBUG");  
        }  
  
        //...  
    }  
}
```

- Framework (biblioteka) służący do logowania w trakcie działania aplikacji
- Napisany w Javie i rozpowszechniany na licencji Apache Software License
- Został przeportowany na C/C++, C#, Python, Ruby, Perl i Eiffel
- Konfigurowalny poprzez zewnętrzne pliki konfiguracyjne w trakcie działania aplikacji
- Wyświetla proces rejestracji pod kątem poziomów (log level)
- Oferuje mechanizmy logowania do wielu różnych miejsc docelowych, np. Baza danych, plik, konsola, etc.

Przykład

HelloWorld.java

```
import org.apache.log4j.Logger;

public class HelloWorld{

    static Logger log = Logger.getLogger(HelloWorld.class.getName());

    public static void main(String[] args) {
        try {
            log.info("Sample info message");
            log.debug("Sample debug message");
        }
        catch(Exception e) {
            log.error("Sample error message", e);
            log.fatal("Sample fatal message ");
        }
    }
}
```

Przykład

Log4j.properties

```
# Define the root logger with appender file
log = /usr/home/log4j
log4j.rootLogger=INFO, FILE, CONSOLE

# Define the file appender
log4j.appender.FILE=org.apache.log4j.FileAppender
log4j.appender.FILE.File=${log}/console.log
log4j.appender.FILE.MaxFileSize=4MB

log4j.appender.CONSOLE=org.apache.log4j.ConsoleAppender

# Define the layout for file appender
log4j.appender.FILE.layout=org.apache.log4j.PatternLayout
log4j.appender.FILE.layout.conversionPattern=%m%n

log4j.appender.CONSOLE.layout.ConversionPattern=%d{yyyy-MM-dd HH:mm:ss.SSS} %-5p (%t) %m\n
```



Narzędzia do logowania

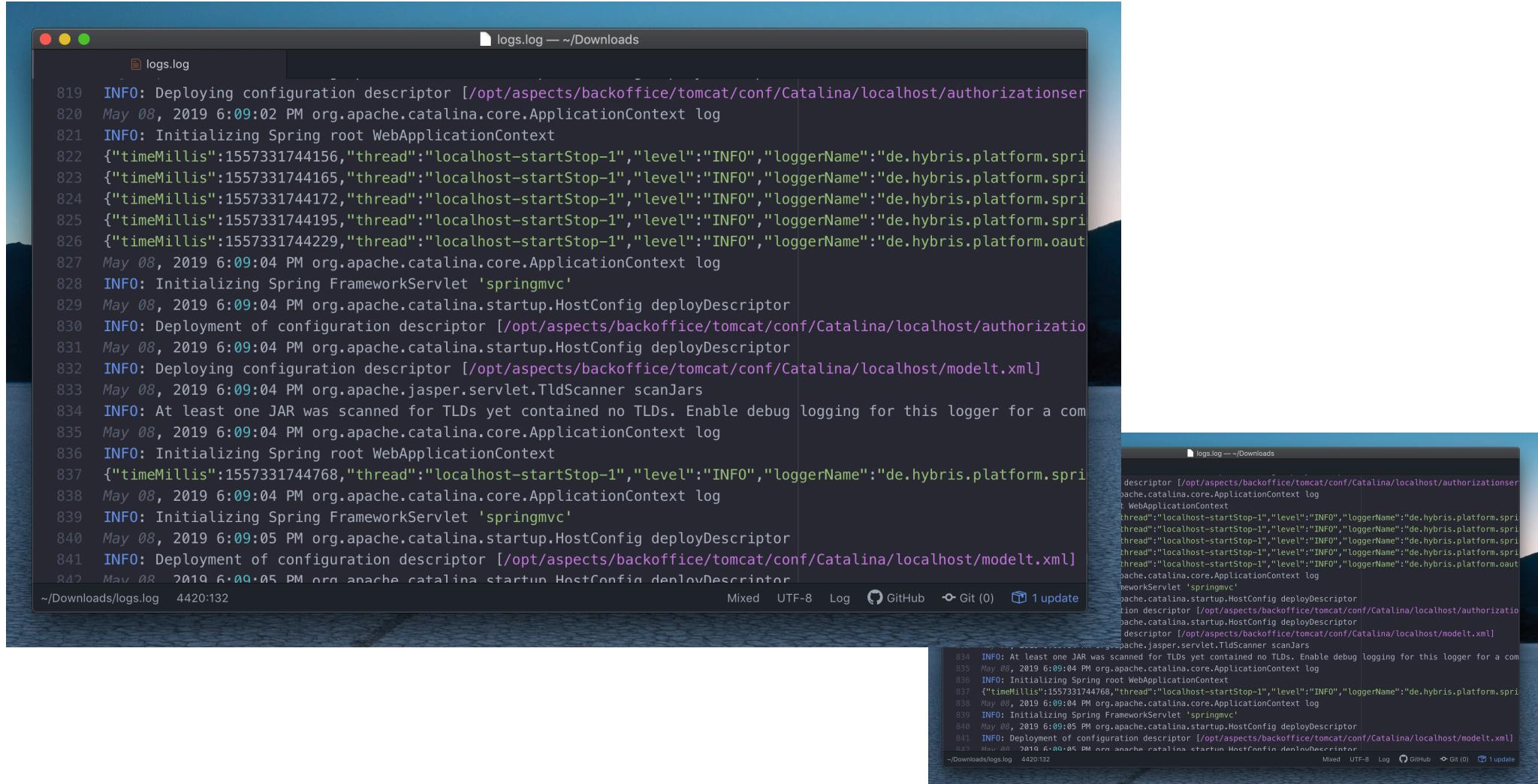
Dobre praktyki logowania



Wytyczne dotyczące polityki logowania w projekcie



Podwójne logi, na różnych poziomach



The image shows two terminal windows side-by-side, both displaying the file `logs.log` from the directory `~/Downloads`. The left window shows log entries from line 819 to 842, while the right window shows entries from line 834 to 888. Both logs are timestamped with "May 08, 2019 6:09:02 PM" or "May 08, 2019 6:09:04 PM". The logs are primarily composed of INFO-level messages from various org.apache.* and de.hybris.* packages, related to the deployment and initialization of configuration descriptors and Spring FrameworkServlets.

```
logs.log — ~/Downloads
logs.log — ~/Downloads

819 INFO: Deploying configuration descriptor [/opt/aspects/backoffice/tomcat/conf/Catalina/localhost/authorizationse
820 May 08, 2019 6:09:02 PM org.apache.catalina.core.ApplicationContext log
821 INFO: Initializing Spring root WebApplicationContext
822 {"timeMillis":1557331744156,"thread":"localhost-startStop-1","level":"INFO","loggerName":"de.hybris.platform.spri
823 {"timeMillis":1557331744165,"thread":"localhost-startStop-1","level":"INFO","loggerName":"de.hybris.platform.spri
824 {"timeMillis":1557331744172,"thread":"localhost-startStop-1","level":"INFO","loggerName":"de.hybris.platform.spri
825 {"timeMillis":1557331744195,"thread":"localhost-startStop-1","level":"INFO","loggerName":"de.hybris.platform.spri
826 {"timeMillis":1557331744229,"thread":"localhost-startStop-1","level":"INFO","loggerName":"de.hybris.platform.oaut
827 May 08, 2019 6:09:04 PM org.apache.catalina.core.ApplicationContext log
828 INFO: Initializing Spring FrameworkServlet 'springmvc'
829 May 08, 2019 6:09:04 PM org.apache.catalina.startup.HostConfig deployDescriptor
830 INFO: Deployment of configuration descriptor [/opt/aspects/backoffice/tomcat/conf/Catalina/localhost/authorizatio
831 May 08, 2019 6:09:04 PM org.apache.catalina.startup.HostConfig deployDescriptor
832 INFO: Deploying configuration descriptor [/opt/aspects/backoffice/tomcat/conf/Catalina/localhost/modelt.xml]
833 May 08, 2019 6:09:04 PM org.apache.jasper.servlet.TldScanner scanJars
834 INFO: At least one JAR was scanned for TLDs yet contained no TLDs. Enable debug logging for this logger for a com
835 May 08, 2019 6:09:04 PM org.apache.catalina.core.ApplicationContext log
836 INFO: Initializing Spring root WebApplicationContext
837 {"timeMillis":1557331744768,"thread":"localhost-startStop-1","level":"INFO","loggerName":"de.hybris.platform.spri
838 May 08, 2019 6:09:04 PM org.apache.catalina.core.ApplicationContext log
839 INFO: Initializing Spring FrameworkServlet 'springmvc'
840 May 08, 2019 6:09:05 PM org.apache.catalina.startup.HostConfig deployDescriptor
841 INFO: Deployment of configuration descriptor [/opt/aspects/backoffice/tomcat/conf/Catalina/localhost/modelt.xml]
842 May 08, 2019 6:09:05 PM org.apache.catalina.startup.HostConfig deployDescriptor
~/Downloads/logs.log 4420:132

834 INFO: At least one JAR was scanned for TLDs yet contained no TLDs. Enable debug logging for this logger for a com
835 May 08, 2019 6:09:04 PM org.apache.catalina.core.ApplicationContext log
836 INFO: Initializing Spring root WebApplicationContext
837 {"timeMillis":1557331744768,"thread":"localhost-startStop-1","level":"INFO","loggerName":"de.hybris.platform.spri
838 May 08, 2019 6:09:04 PM org.apache.catalina.core.ApplicationContext log
839 INFO: Initializing Spring FrameworkServlet 'springmvc'
840 May 08, 2019 6:09:05 PM org.apache.catalina.startup.HostConfig deployDescriptor
841 INFO: Deployment of configuration descriptor [/opt/aspects/backoffice/tomcat/conf/Catalina/localhost/modelt.xml]
842 May 08, 2019 6:09:05 PM org.apache.catalina.startup.HostConfig deployDescriptor
~/Downloads/logs.log 4420:132
```

Logowanie wybranych informacji, a nie całych obiektów

```
employee.EmployeeController : User 3called  
employee.EmployeeController : Employee(id=3, firstName=Bilbo, lastName=Baggins, role=burlar)
```

```
log.info("User " + employee.getId() + " called");  
log.info(employee.toString());
```



Antywzorzec *if (logger.isXXXEnabled())*

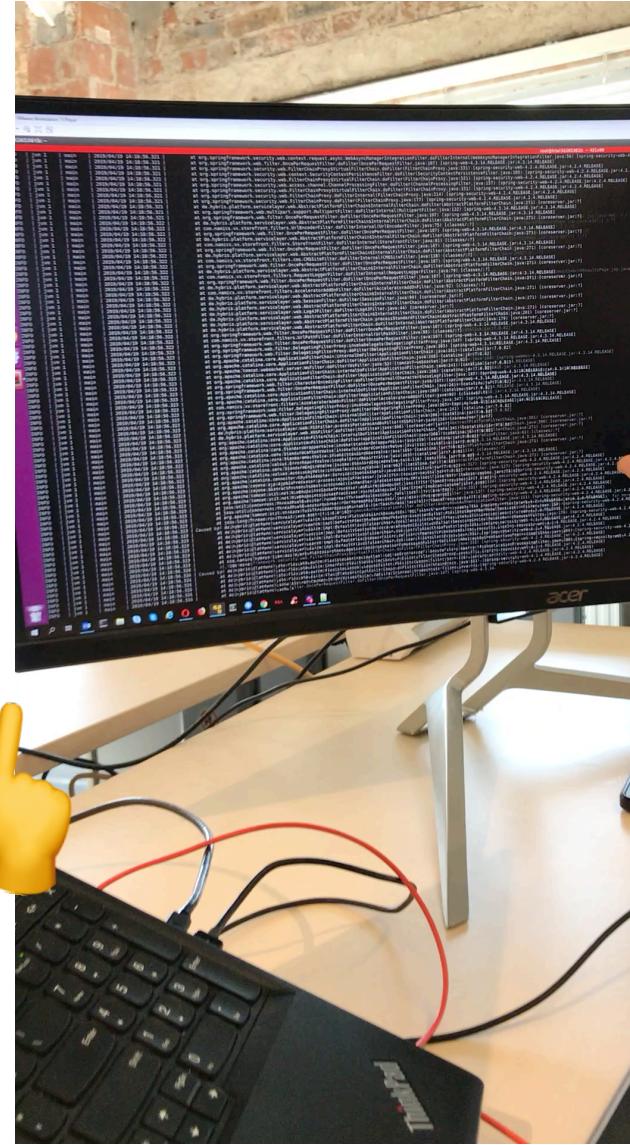
```
log.debug("Hey ho: " + orderRepository);  
}
```

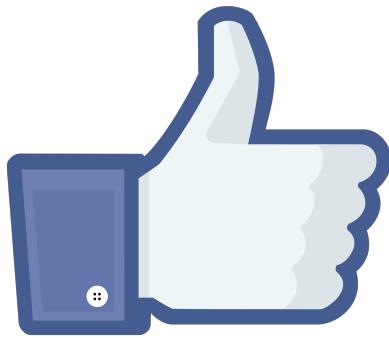
Lombok

```
11 import org.springframework.context.annotation.Configuration,  
12  
13 @Configuration  
14 @Slf4j  
15 public class LoadDatabase {  
16  
17     @Bean  
18     CommandLineRunner initDatabase (EmployeeRepository repository, OrderRepository orderRepository) {  
19         return args -> {  
20             orderRepository.save(new Order(description: "MacBook Pro", Status.COMPLETED));  
21             orderRepository.save(new Order(description: "iPhone", Status.IN_PROGRESS));  
22  
23             orderRepository.findAll().forEach(order -> {  
24                 log.info("Preloaded " + order);  
25             });  
26             log.info("Preloading " + repository.save(new Employee(name: "Bilbo Baggins", role: "burlar")));  
27             log.info("Preloading " + repository.save(new Employee(name: "Frodo Baggins", role: "thief")));  
28  
29         };  
30     }  
31 }  
32 }
```

Zbyt duża szczegółowość

Case study!





- Ułatwia debugowanie oraz utrzymanie aplikacji 🎉😊
- Umożliwia uporządkowane przechowanie informacji o działającej aplikacji
- Może spowolnić aplikację
- Zbyt szczegółowy może prowadzić do oczopłasu 😳...
- ... i do wyczerpania miejsca na dysku

Podsumujmy



O czym należy pamiętać

- Wspólne dla całego projektu/firmy zasady logowania 
- Podział na pliki pod względem szczegółowości 
- Uważanie na to aby nie wyciekły żadne poufne informacje 
- Dopilnowanie aby pliki z logami nie były za duże 

Szczegółowość logów



Console.log size **354 KB**

VS

Console.log size **3.54 GB**

Dobra, więc jak
ogarnąć logowanie?



Jak zarządzać logami?



tail | grep

grep – komenda wyszukująca i wypisującą linie pasujące do podanego wzorca

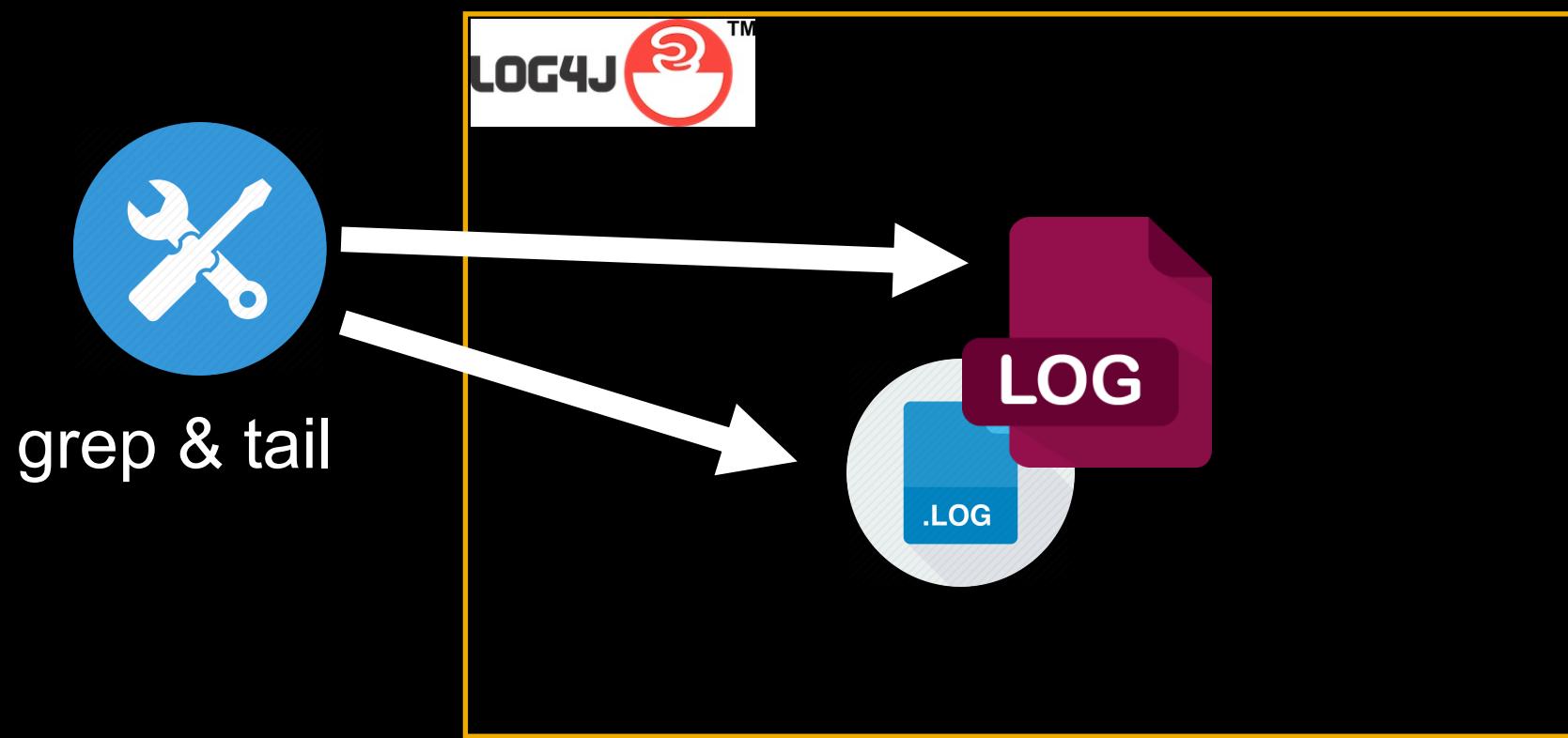
- `grep "DEBUG" console.log > log.log // kopiuje do nowego pliku`
- `grep -c "ERROR" * // ilość trafień`
- `grep -v "ERROR\|DEBUG" console.log | grep "10:56:03.4" // negacja + kombinacja`
- `grep -Eoc "^[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}.* 404" access.log`

tail | grep

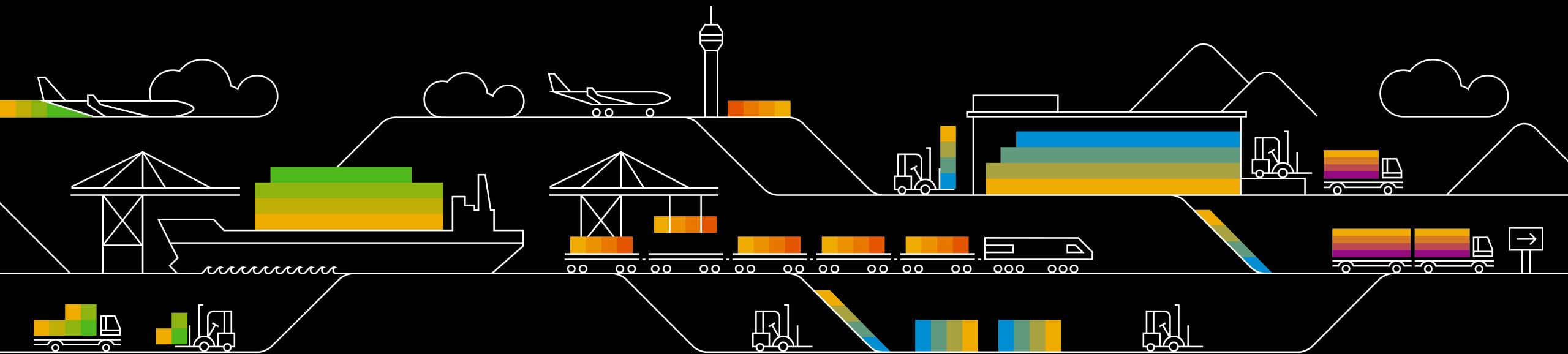
tail – komenda wypisująca podaną liczbę linii pliku

- tail -n 100 console-1
- tail -f console
- tail -n 200 console-2





Narzędzia do zarządzania logami



Przykładowe narzędzia



Prometheus



Grafana



GROW



kibana



GECKOBOARD



Segment

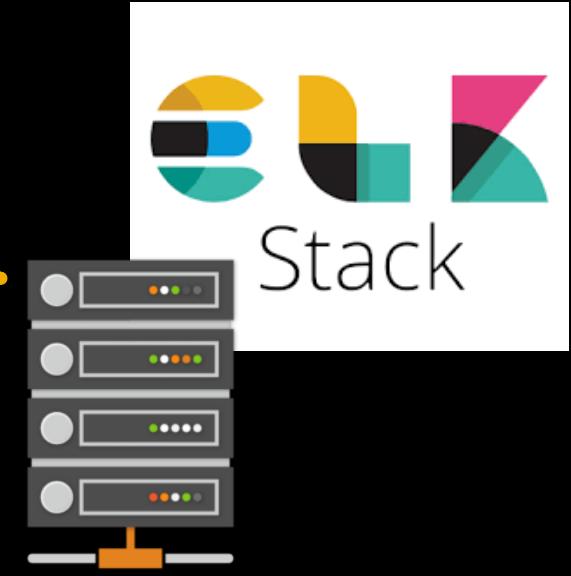
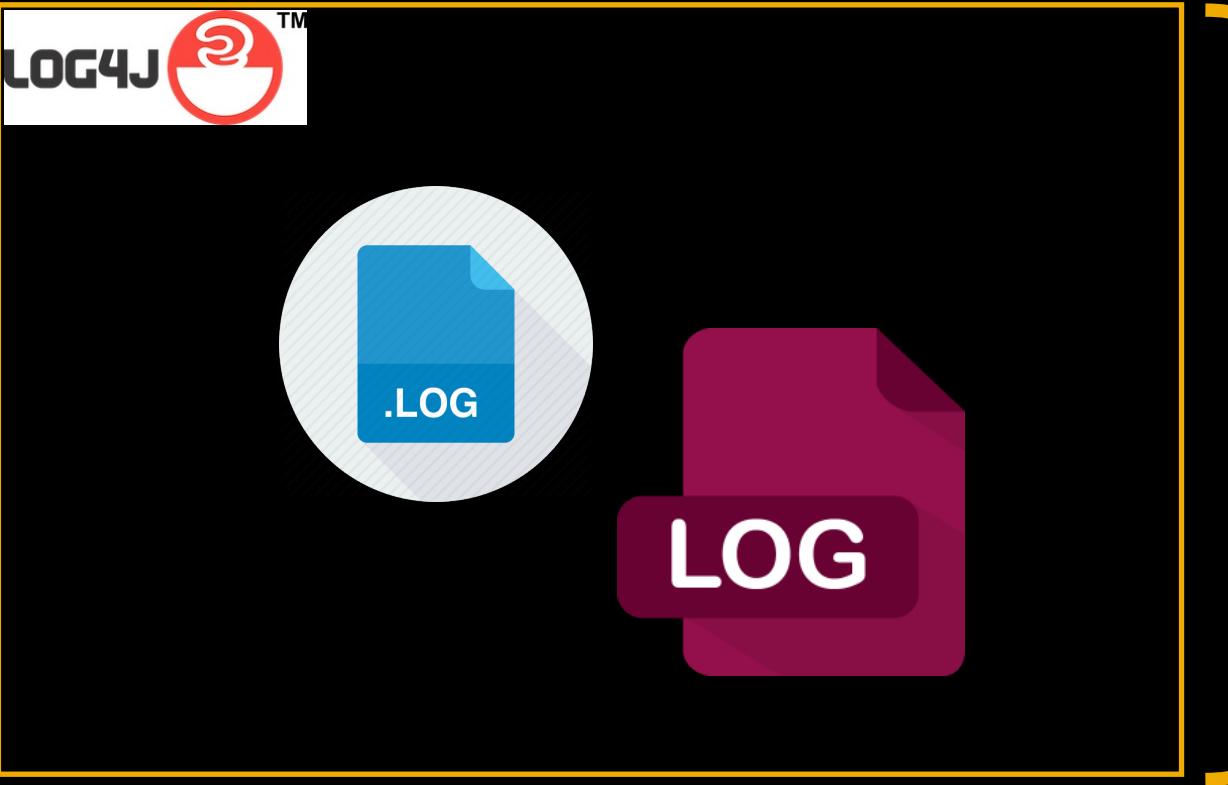
splunk®

Czym jest ELK?





grep & tail



Słowo na koniec



Źródła

Kobiety do kodu: Praktyczna Java. Biblioteki do logowania

<https://kobietydokodu.pl/praktyczna-java-biblioteki-do-logowania/>

Spring Boot Logs with Elasticsearch, Logstash and Kibana(ELK)

<https://medium.com/hprog99/spring-boot-logs-with-elasticsearch-logstash-and-kibana-elk-c61a378f3cb4>

Thank you.

Contact information:

Henryk Nowakowski henryk.nowakowski@sap.com

Sebastian Rulik sebastian.rulik@sap.com