

Week 1 Unit 1

- 00:00:05 Hola y bienvenido a la primera semana, introducción del modelo de programación de aplicaciones RESTful ABAP. En esta unidad, la primera unidad, vamos a hablar del panorama general del RAP.
- 00:00:12 Entonces, ¿qué es RAP? Si desea comprender lo que RAP tiene que soportar, primero tenemos que mirar las aplicaciones.
- 00:00:20 Entonces, ¿qué deben admitir las aplicaciones? La experiencia de usuario es realmente importante.
- 00:00:24 Anteriormente creábamos transacciones ABAP realmente grandes, que ofrecían una gran cantidad de funcionalidad a los usuarios finales. El usuario final decide qué funcionalidad es realmente importante.
- 00:00:32 Pero en el mundo moderno, esto tiene que cambiar. Queremos dar soporte al usuario final y a los negocios diarios y ofrecer la funcionalidad requerida.
- 00:00:40 Esto también mejora la productividad. Por lo tanto, las aplicaciones basadas en roles y específicas de cada rol son realmente importantes.
- 00:00:47 Deben ejecutarse en el navegador, no queremos instalar ningún cliente específico. Y, por supuesto, también deberían funcionar en cualquier dispositivo, y también sería deseable un cambio de dispositivo.
- 00:00:56 SAP Fiori y SAPUI5 admiten pedidos basados en los datos de protocolo RESTful. Pero el rendimiento también es crucial para la experiencia del usuario.
- 00:01:05 Con nuestra base de datos in-memory, SAP HANA, podemos ofrecer nuevas características y mejorar el rendimiento significativamente. HANA también ofrece la combinación de capacidades de transacción con capacidades analíticas, y se pueden integrar en una aplicación
- 00:01:21 Fiori. Además de la experiencia del usuario, también la experiencia de desarrollo es realmente interesante.
- 00:01:26 El usuario final debería divertirse al utilizar la aplicación y el desarrollador de aplicaciones debería tener la posibilidad de divertirse al crear la aplicación.
- 00:01:32 Por lo tanto, la probabilidad, el soporte y la documentabilidad son ciudadanos de primera clase. El desarrollo seguro de aplicaciones y la alta calidad de código suelen ser compatibles con las herramientas ABAP.
- 00:01:43 Pero, ¿qué sucede ahora si vengo del viejo mundo heredado y quiero pasar al futuro, a la nube? La migración de códigos personalizados admite esto, podemos pasar al nuevo mundo.
- 00:01:53 En la nube, los nuevos requisitos están en la cima. Queremos definir las aplicaciones que se pueden ampliar.
- 00:01:59 Por lo tanto, creamos anclas estables que sean compatibles con el futuro. Y si desea crear nuevas aplicaciones, aplicaciones más grandes, podemos utilizar el escenario paralelo compatible con SAP Cloud
- 00:02:09 Platform como plataforma de extensión. API en la lista blanca para que se pueda utilizar cualquier back end.
- 00:02:15 Sin estado es crucial para todas estas aplicaciones, porque solo con eso podemos hacer uso de la elasticidad y escalabilidad en la nube. Pero, ¿cuál es la misión de eso?

00:02:27 No introduciremos un nuevo modelo de programa cada cinco años. El RAP es una oportunidad única en toda la vida porque tenemos a todos los actores clave.

00:02:35 ADT, los idiomas y también los marcos. Por lo tanto, RAP es una solución estratégica a largo plazo por nuestra parte.

00:02:43 Podemos usarlo para el desarrollo eficiente de las aplicaciones Fiori y también de las API web. Esto se puede hacer para aplicaciones creadas desde cero o integrando el código de aplicación existente.

00:02:58 Queremos ofrecer una experiencia de desarrollo integral con un flujo de desarrollo estandarizado basado en mejores prácticas y recortes de desarrollo.

00:03:05 La alta eficiencia de desarrollo es realmente crucial para nosotros y, por lo tanto, el desarrollador debe enfocarse en la lógica de negocio y la funcionalidad de negocio.

00:03:12 Y asumimos todas las tareas técnicas de implementación. Como he dicho, la probabilidad, la documentabilidad y el soporte son ciudadanos de primera clase e integrados de forma nativa.

00:03:21 Lo mismo ocurre con la transferencia de código a SAP HANA, por supuesto. El núcleo de la aplicación también debe sobrevivir en un mundo sin estado y con estado.

00:03:30 Cuatro cualidades de producto con SAP Fiori y SAP HANA también apuntan al entorno en la nube, como ya he introducido. La extensibilidad es un factor realmente crucial para nosotros, que también podemos verticalizar las aplicaciones.

00:03:44 Pero, ¿cuáles son ahora los actores clave de los RAP? Como he dicho, ADT es, por supuesto, una herramienta de elección para nosotros.

00:03:52 ADT, las herramientas de desarrollo ABAP dan soporte a una incorporación fácil de desarrolladores con un flujo de desarrollo integral.

00:03:58 Por lo tanto, debería poder crear nuevas aplicaciones sin crear toneladas de documentación. El lenguaje ABAP y ABAP CDS también deben admitir las tareas de desarrollo, las tareas de implementación estándar o, según sea

00:04:10 necesario, las API tipificadas en RAP. No queremos tener API generadas o API genéricas.

00:04:16 Las verificaciones de código estático, la finalización automática y la información de elementos son puntos cruciales para mejorar la productividad del desarrollo.

00:04:24 Los marcos potentes asumen las tareas de implementación técnica para que podamos centrarnos en las empresas. Pero, ¿qué se puede apoyar ahora con RAP?

00:04:34 Podemos desarrollar servicios para las aplicaciones SAP Fiori y para las API web, podemos hacerlo en base al código heredado o en un enfoque nuevo, y podemos implementar estas aplicaciones para todas las diferentes líneas de código.

00:04:47 RAP es solo una línea de código para todas las diferentes opciones de implementación, SAP S/4HANA o los escenarios en la nube. Si echamos un vistazo a la pila de extremo a extremo, podemos ver que todo se basa en CDS.

00:04:59 Y el núcleo de RAP, basado en CDS, admite todas las funciones de borrador, la extensibilidad y la implementación lista para ser utilizada para llevar a cabo todas las tareas técnicas de implementación.

00:05:11 OData es un protocolo que ofrece muchas funcionalidades, como una consulta potente, y con SAP Fiori y UI5, podemos crear buenas aplicaciones Fiori.

00:05:21 El consumo de servicios a través de API web se puede integrar en cualquier otro cliente. Pero, ¿qué significa para nuestros clientes ahora?

00:05:29 Algunos clientes aún están en una versión ABAP inferior a 7.5, donde se incluyen muchas tecnologías y las mejores prácticas para aplicarlas.

00:05:36 Pero con 7.5, fue la primera vez que introdujimos un modelo de programación integral. Desde CDS mediante BOPF hasta el nivel de gateway, pudimos crear aplicaciones.

00:05:48 Pero teníamos algunos obstáculos y algunas deficiencias, por lo que presentamos el sucesor del modelo de programación ABAP para SAP Fiori.

- 00:05:54 Por lo tanto, RAP es el sucesor de este. Ampliamos los servicios de datos centrales mediante la denominada definición de comportamiento.
- 00:06:02 La extensión del lenguaje CDS, y también la extensión del propio lenguaje ABAP, se utilizan para tener un concepto nativo de business objects integrados.
- 00:06:12 Y el uso de CDS, así como los business objects, se pueden exponer mediante servicios empresariales, completamente integrados con ADT. Pero, ¿qué significa ahora que ya he invertido en el modelo de programación ABAP para SAP Fiori?
- 00:06:24 Por lo tanto, no se preocupe, nos aseguraremos de que pronto habrá una integración en funcionamiento. En esta unidad, aprendimos sobre los requisitos de las aplicaciones, desde el punto de vista de la experiencia del usuario y también
- 00:06:34 desde el punto de vista de la experiencia de desarrollo. Entendimos la misión de RAP y entendimos que el RAP es la solución estratégica a largo plazo.
- 00:06:42 En la siguiente unidad, veremos con más detenimiento el panorama arquitectónico. Así que estaré encantado de verte ahí.

Week 1 Unit 2

- 00:00:05 Hola, y bienvenido a la primera semana, introducción del modelo de programación de aplicaciones ABAP RESTful. En la segunda unidad, hablaremos del resumen arquitectónico.
- 00:00:12 RAP tiene básicamente tres capas. Las entidades de CD y los business objects se definen en el modelado de datos y en la capa de comportamiento.
- 00:00:21 Hemos analizado que queremos crear aplicaciones específicas de rol en la primera unidad. Por lo tanto, sobre la base de nuestro modelo de datos, la funcionalidad empresarial contenida en nuestro objeto empresarial puede proyectarse de forma
- 00:00:34 específica para cada función. Estas proyecciones de business object, o proyecciones CDS, se pueden exponer mediante una definición de servicio.
- 00:00:42 Por lo tanto, una definición de servicio es un paréntesis alrededor de este contenido y define el alcance que se expondrá en la vinculación de servicios. La vinculación de servicio define el protocolo y también el caso de utilización.
- 00:00:53 API web, por ejemplo, de su dispositivo. Pero, ¿cuál es exactamente nuestro objeto de negocio?
- 00:00:59 El business object consta básicamente de tres partes. La primera parte es el modelo de datos.
- 00:01:06 La entidad CDS define la raíz y, mediante las composiciones, podemos navegar a los nodos subordinados. Por lo tanto, el business object, como el interlocutor comercial, el producto o el pedido de cliente con todos los nodos subordinados, es básicamente un árbol de
- 00:01:22 composición. Además de este modelo de datos, ahora puede adjuntar un comportamiento.
- 00:01:28 Y esto convierte ahora los artefactos CDS en un business object, ampliándolo mediante una definición de comportamiento. Podemos especificar operaciones por defecto u operaciones estándar como crear, leer, actualizar y borrar, así como operaciones específicas como acciones y
- 00:01:42 funciones. La gestión de la concurrencia se realiza mediante bloqueos y etiquetas electrónicas.
- 00:01:46 Las autorizaciones, el control de características y el borrador también son funciones muy importantes. Y ahora la tercera parte del business object se define mediante el tiempo de ejecución o la implementación.
- 00:01:58 Independientemente de si estamos utilizando o reutilizando el código de aplicación existente o iniciando o empezando desde cero, el objeto empresarial también tiene tres partes importantes.
- 00:02:06 La primera parte es la fase de interacción. Cuando el usuario final pulsa el botón Editar en una aplicación, hasta que presione el botón Guardar, todos los datos que se introdujeron mientras tanto se
- 00:02:17 escriben en una memoria intermedia transaccional. La memoria intermedia transaccional se escribe en la secuencia de grabación en la base de datos.
- 00:02:25 Ahora, el almacenamiento de todas las modificaciones en una memoria intermedia transaccional, por supuesto, no funciona para una aplicación sin estado y, como hablamos en la primera unidad, la comunicación sin estado es realmente crucial para nuestro entorno en la nube.
- 00:02:38 Así que el borrador llena ahora la brecha entre la comunicación sin estado necesaria para el entorno de la nube y la naturaleza de una aplicación con estado, porque en algún lugar se deben almacenar los cambios.
- 00:02:50 El borrador conserva ahora la memoria intermedia transaccional o las modificaciones escritas en la memoria intermedia transaccional en la base de datos.
- 00:02:57 Por lo tanto, el borrador se convierte en su propio recurso, al que también puede acceder OData. Y, por lo tanto, el proyecto llena el vacío.

- 00:03:05 No tenemos sesiones ABAP entre dos llamadas, solo tenemos una sesión ABAP mientras ejecutamos los cambios. Por supuesto, las modificaciones dentro de un borrador también se almacenan en una memoria intermedia transaccional antes de que se graben de forma persistente.
- 00:03:19 Y ahora el borrador también requiere, por supuesto, bloqueos, por lo que ampliamos la funcionalidad del servidor ABAP mediante bloqueos en cola. Los bloqueos de cola ya no están vinculados a una sesión ABAP, sino que están vinculados al propio borrador, la instancia de borrador en la base de datos.
- 00:03:34 Esto significa en detalle que la fase de interacción contiene todas las llamadas modificadas, las llamadas de lectura y los bloqueos. Las modificaciones se escriben en la memoria intermedia transaccional y la secuencia de guardado solo para los datos activos, porque para el borrador no
- 00:03:47 tiene una secuencia de guardado. Para los datos activos, hemos finalizado, `check_before_save`, `ajústque_números` y grabamos.
- 00:03:54 Al finalizar, todos los business objects pueden realizar las modificaciones más recientes. ¿Por qué?
- 00:03:59 Porque solo si ya no se pueden aplicar modificaciones, cada business object implicado en una transacción o aplicación puede decidir, en `check_before_save`, si el estado actual es consistente o no.
- 00:04:11 Y si cada business object está de acuerdo en que el estado actual es consistente, podemos desencadenar los `ajust_number` para llamar los números, por motivos legales, muy tarde, y también puede persistir las modificaciones mediante el modelo de función de tarea de actualización.
- 00:04:26 El número de ajuste es, por supuesto, un paso opcional. Y ahora, si tiene un montón de código existente, puede hacer uso de un business object con un tiempo de ejecución no gestionado.
- 00:04:36 Esto significa que no gestionamos el tiempo de ejecución completamente, pero tiene la memoria intermedia transaccional, implementa, crea, lee, actualiza el lead y las acciones y funciones completamente, porque ya tiene disponible su lógica empresarial.
- 00:04:48 Ya estaba allí antes de pensar en las aplicaciones RAP y Fiori, tal vez. Se realizó porque desvinculó su codificación de los módulos Dynpro cuando también quería crear BAPI o integrar su lógica en una aplicación Web Dynpro ABAP.
- 00:05:05 Definir adaptadores es una inversión sostenible una vez y nos ocupamos de una mayor integración en gateway, OData V2, OData V4 o incluso otros protocolos. Pero también hablamos de que, además del punto crucial de integrar la lógica existente, también es muy importante que podamos crear nuevas aplicaciones desde
- 00:05:25 cero para que RAP pueda hacerse cargo de las tareas de implementación técnica, y el desarrollador de aplicaciones pueda enfocarse más en la lógica de negocio en sí.
- 00:05:34 Por lo tanto, también ofrecemos el tipo de implementación gestionado. En la gestión, las operaciones estándar están listas para usar.
- 00:05:42 Sin una sola línea de código, podemos tener una interfaz de usuario completa que admita crear, leer, actualizar, eliminar, con toda la funcionalidad de borrador. Y ahora el desarrollador de la aplicación puede adjuntar lógica empresarial como determinaciones y validaciones, y lógica específica para el control de funciones
- 00:05:59 y autorizaciones. Por supuesto, aquí también admitimos los breakouts, por ejemplo, puede reutilizar los objetos de bloqueo existentes o los módulos de funciones de tarea de
- 00:06:09 actualización existentes, por lo que puede elegir libremente dónde debemos encargarse de sus tareas de implementación técnica y dónde necesita la libertad para implementar lo que desea hacer usted mismo.
- 00:06:20 Entonces, ¿cómo se ve el flujo de desarrollo? En primer lugar, comenzamos con CDS, la primera parte de un business object.

- 00:06:28 Y en el caso de un business object, podemos ampliar el modelo de datos mediante la definición de comportamiento e implementarlo mediante código ABAP. Y ahora se basa completamente en el tipo de implementación, ya sea que implementemos crear, leer, actualizar, eliminar o si implementamos
- 00:06:44 determinaciones y validaciones. Un objeto de bloqueo se puede integrar si existe o se utiliza el objeto de bloqueo gestionado genérico.
- 00:06:55 Esto también se puede incluir, por supuesto, en el objeto de autorización, al que se puede acceder mediante un DCL en CDS. La prueba de unidad también es posible para nuestras unidades muy pequeñas en el objeto de negocio, y también se pueden escribir pruebas de integración basadas en
- 00:07:10 nuestro objeto de negocio a través del Lenguaje de manipulación de entidades. EML es la API ABAP integrada y documentada estandarizada para acceder a cualquier tipo de objeto empresarial desde fuera.
- 00:07:25 Entonces, ¿cómo puedo crear servicios empresariales en la parte superior, es decir, aplicaciones específicas de rol, por ejemplo? Entonces, en primer lugar, ¿qué es un servicio empresarial?
- 00:07:34 Un servicio empresarial se define básicamente mediante una definición de servicio. Como he dicho, define los paréntesis alrededor de las entidades que se van a exponer.
- 00:07:44 La proyección define qué elementos se deben exponer mediante CDS. La definición de comportamiento define qué comportamiento debe exponerse.
- 00:07:54 Como he dicho, tenemos el objeto empresarial, que se puede reutilizar en varias aplicaciones. Definimos una proyección en la parte superior, qué alcance es realmente relevante y, a continuación, la definición de servicio define qué diferentes partes del
- 00:08:08 business object, incluso varios business objects, y cuáles de ellos se exponen a una vinculación de servicio, que está en la segunda parte. La vinculación de servicio define si es OData V2 o V4 y qué caso de uso se debe admitir.
- 00:08:25 En el caso de las API web, la liberación, la creación de versiones y la obsolescencia son realmente importantes. Porque si libero una API web, podemos ofrecer verificaciones de compatibilidad que las API de RAP sean competitivas para el futuro.
- 00:08:40 Pero si queremos tener unos ingresos planificados que cambien, también necesitamos algún tipo de creación de versiones. Para que la nueva versión incluya la innovación y la antigua no.
- 00:08:49 Y si tenemos varias versiones, quizás en algún momento, también queremos dejar obsoleta una versión antigua. El soporte de herramientas también es muy importante, y porque si conocemos el escenario definido en el enlace de servicios, podemos distinguir las
- 00:09:02 herramientas. Por lo tanto, para la IU, podemos ofrecer una vista previa de los elementos de Fiori.
- 00:09:06 Para API web, una vista previa de IU Swagger puede ser útil. Ahora, es en la naturaleza de un servicio OData de Fiori o API web que no todas las funciones son necesarias, y queremos optimizar el servicio OData a cuál es el
- 00:09:18 escenario específico? Por lo tanto, es muy importante que comprendamos para qué escenario se utiliza el servicio OData.
- 00:09:26 Las anotaciones de IU solo son relevantes para IU, la misma para la ayuda para entradas y el borrador y el control futuro dinámico. Por otro lado, las API web requieren solicitudes más complejas.
- 00:09:38 Solicitudes por lotes, tal vez también que contengan IDs de contenido, donde defina un nuevo documento o una nueva instancia, y después quiero ejecutar operaciones basadas en esta instancia dentro de una solicitud \$batch.
- 00:09:52 Este es un ID de contenido que hace referencia a. La liberación, la creación de versiones y la obsolescencia pueden ser también más relevantes para las API web.

- 00:10:03 El flujo de desarrollo ahora se amplía creando la proyección. Y luego, como he dicho, definir la definición del servicio.
- 00:10:12 En el enlace de servicio, podemos ver la vista previa. Lo que también es muy importante es que asignemos los roles para un servicio específico; de lo contrario, no podemos ejecutarlo.
- 00:10:23 La prueba ahora también es compatible con la caja de cliente local de la puerta de enlace. Podemos ejecutar tests de unidad locales sin HTTP.
- 00:10:32 Esto es realmente útil porque podemos hacer uso de todos los diferentes marcos de tabla de prueba que están disponibles, porque esto solo es factible si tenemos que ejecutarlo en una sesión ABAP y no a través de HTTP, por lo que es bueno.
- 00:10:47 En esta sesión, aprendió sobre la arquitectura del modelo de programación de aplicaciones ABAP RESTful y las partes vitales. Aprendió sobre business objects y servicios empresariales y cómo todo esto encaja.
- 00:10:58 Aprendió sobre los objetos de negocio no gestionados y gestionados y por qué los ofrecemos. Uno es para el código heredado y otro para asumir tareas de implementación técnica para probar la eficiencia del desarrollo.
- 00:11:11 Y usted vio el flujo de desarrollo de las aplicaciones RAP. Y adivinen qué, exactamente este flujo de desarrollo también estaba respaldado por nuestra herramienta IDT, que siempre estará encantada de navegar de un paso
- 00:11:22 al siguiente haciendo clic con el botón derecho en el menú contextual. En la siguiente unidad, aprenderá sobre las tecnologías involucradas.
- 00:11:29 Nos vemos allí.

Week 1 Unit 3

- 00:00:05 Hola, y bienvenidos hasta la semana uno, unidad tres del curso de openSAP Creación de aplicaciones con el modelo de programación de aplicaciones RESTful ABAP.
- 00:00:13 Mi nombre es Carine Tchoutouo Djomo y soy un gestor de productos para la plataforma ABAP. Estoy encantado de guiarle a través de las próximas unidades de esta semana.
- 00:00:22 En unidades anteriores, ha aprendido sobre el panorama general y la arquitectura del modelo de programación de aplicaciones RESTful ABAP, en resumen, RAP.
- 00:00:31 Ahora vamos a echar un vistazo a las diversas tecnologías e instrumentos implicados en este contexto.
- 00:00:37 Entrémonos en ello. La plataforma ABAP ha sido fuertemente optimizada para trabajar con SAP HANA como la plataforma de
- 00:00:49 base de datos subyacente en los últimos años. SAP HANA es la plataforma de base de datos subyacente al núcleo digital de SAP, SAP S/4HANA y
- 00:00:59 la oferta de la plataforma ABAP como servicio de SAP, el entorno ABAP de SAP Cloud Platform. Con sus aspectos tecnológicos clave, como computación in-memory, almacenamiento de datos
- 00:01:12 basado en filas y en columnas, alta compresión de datos y soporte de arquitectura multinúcleo, SAP HANA ha tenido un gran impacto en el desarrollo de aplicaciones ABAP.
- 00:01:22 Las mejoras clave aquí son, en primer lugar, el nuevo paradigma de programación en la plataforma ABAP llamado inserción de código, donde los cálculos intensivos en datos se delegan a la capa
- 00:01:34 de base de datos en lugar de procesarse en la capa de aplicación. La base de datos ya no es un cuello de botella.
- 00:01:43 Los agregados materializados ya no son necesarios, ya que se pueden calcular sobre la marcha y con mayor flexibilidad gracias a la computación in-memory de alta velocidad.
- 00:01:55 Se necesitan menos índices gracias al almacén de datos basado en columnas, que permite una rápida recuperación de datos de columnas en conjuntos de datos.
- 00:02:06 Sin necesidad de agregados e índices, se reducen las líneas de código y la complejidad de los modelos de datos.
- 00:02:18 La plataforma ABAP ofrece diferentes técnicas de programación pushdown de código para aprovechar fácilmente la potencia de SAP HANA en aplicaciones basadas en ABAP.
- 00:02:28 Las principales técnicas de programación de inserción de código gestionado por ABAP son ABAP Core Data Services, en resumen, CDS y ABAP SQL, antes conocido como Open SQL.
- 00:02:41 La mayoría de los escenarios se pueden implementar con ambas técnicas. Para escenarios avanzados con requisitos específicos, o que requieren acceso a
- 00:02:52 funcionalidades nativas de SAP HANA como Predictive Analysis Library, Data Mining y geoespacial, se pueden utilizar las funciones y funciones de la base de datos gestionada por ABAP
- 00:03:08 y las funciones y jerarquías de tablas CDS. CDS es la infraestructura de modelado de datos de próxima generación en la plataforma ABAP y la
- 00:03:18 piedra angular de RAP. CDS ofrece un enfoque declarativo para definir modelos de datos específicos de dominio y ricos
- 00:03:26 semánticamente cerca del pensamiento conceptual, lo que facilita la definición, comprensión y consulta de los modelos de datos.
- 00:03:34 La expresividad en los modelos de datos se logra mediante un amplio conjunto de funciones integradas y funciones de inserción de código, como distinciones de casos, agregaciones y
- 00:03:47 cadenas, aritméticas y funciones de conversión. Las asociaciones CDS se pueden utilizar para definir las relaciones entre entidades

- 00:03:56 empresariales a nivel de modelo de datos, evitando la necesidad de JOINS explícitos en las sentencias de consulta.
- 00:04:04 Se puede añadir semántica específica de dominio adicional a los modelos de datos utilizando anotaciones CDS y ser evaluada por diferentes tiempos de ejecución y marcos para enriquecer su
- 00:04:16 funcionalidad. CDS también viene con su propio idioma de control de datos para un control de acceso declarativo a
- 00:04:26 nivel de modelo de datos. En términos de creación de vistas, SAP HANA ofrece una variedad de funcionalidades avanzadas de
- 00:04:36 modelado de datos, como vistas analíticas, de atributos y calculadas, pero, por desgracia, la plataforma ABAP no pudo hacer el uso completo de ellas debido a restricciones técnicas.
- 00:04:48 Por otro lado, la plataforma ABAP proporciona una gestión del ciclo de vida probada y potente y tiene su propio sistema de tipo, mientras que SAP HANA viene con su propio sistema de tipo y gestión
- 00:05:03 del ciclo de vida. Con ABAP CDS, los desarrolladores de ABAP ahora pueden utilizar fácilmente la vista SAP HANA de
- 00:05:11 vanguardia para crear capacidades en sus aplicaciones. Esto se consigue mediante funciones avanzadas de modelado de datos CDS, inserción de código
- 00:05:21 elaborado y desgloses de código. ABAP CDS está completamente integrado en la plataforma ABAP y también solo lo gestiona la
- 00:05:29 plataforma ABAP. CDS se puede consumir en ABAP SQL y en los marcos ABAP.
- 00:05:40 Como puede ver en la pantalla, se admiten varias entidades de modelado CDS en la plataforma ABAP. Las vistas CDS se utilizan para la creación de vistas estándar.
- 00:05:50 Aquí tenemos las entidades CDS, las vistas de proyección CDS y las vistas basadas en Dictionary ABAP CDS.
- 00:05:59 Las entidades de vista CDS son el sucesor de las vistas basadas en diccionarios de datos CDS y están disponibles localmente a partir de la plataforma ABAP 2020.
- 00:06:10 Las definiciones de comportamiento CDS se utilizan para definir el comportamiento transaccional de los objetos empresariales RAP.
- 00:06:18 Un modelo de datos CDS se puede ampliar a nivel de modelo de datos así como a nivel de metadatos de forma libre de modificaciones utilizando la ampliación de vista CDS y las ampliaciones de
- 00:06:31 metadatos CDS. Como ya se ha mencionado, las funciones de tabla CDS y las jerarquías CDS se utilizan para acceder
- 00:06:39 a las capacidades avanzadas de SAP HANA. Las entidades personalizadas CDS se utilizan para modelos de datos cuyo tiempo de ejecución se
- 00:06:47 implementa manualmente. Permiten la puesta a disposición de datos mediante clases ABAP.
- 00:06:54 Una entidad abstracta CDS representa todos los atributos de tipo de una entidad CDS sin crear ninguna otra instancia de objeto en la base de datos o en la pila ABAP.
- 00:07:05 Una entidad CDS abstracta se puede utilizar como tipos de datos cuyos atributos de tipo van más allá de las estructuras normales del Dictionary ABAP.
- 00:07:15 Gracias al lenguaje de control de datos CDS, los roles CDS se pueden definir en entidades CDS ABAP no abstractas para restringir la devolución de datos.
- 00:07:29 Como puede ver en la pantalla, en el panorama general del RAP ya introducido en la unidad dos, CDS desempeña un papel importante en RAP.
- 00:07:38 Se utiliza para el modelado de datos y la definición de comportamiento de consultas, business objects y proyecciones de business object.
- 00:07:46 Como ya se ha explicado en la unidad anterior, ABAP CDS y el lenguaje ABAP son los lenguajes de programación para todas las tareas de implementación RAP estándar.
- 00:08:00 Desde el principio hasta ahora, el lenguaje ABAP se ha modernizado constantemente de varias maneras.

00:08:06	La modernización más reciente se centra en proporcionar un lenguaje de desarrollo declarativo y orientado a la expresión.
00:08:16	Las funciones más importantes que se proporcionan aquí son varias declaraciones en línea y nuevas expresiones de constructor con inferencia de tipo para evitar la definición de variables
00:08:27	innecesarias en su código. Una sintaxis ampliamente orientada a la expresión y también potentes operaciones de tabla internas
00:08:38	como agrupación, filtrado y el operador CORRESPONDING(). Un lenguaje de manipulación de entidad para controlar el comportamiento transaccional de
00:08:52	objetos empresariales en el contexto de RAP. El soporte de la biblioteca JSON en la biblioteca sXML y el soporte para la documentación de código
00:09:04	en línea con ABAP Doc con una opción para exportar los comentarios ABAP Doc. El soporte para la prueba de unidad ABAP con varias técnicas de aislamiento utilizando marcos
00:09:18	de pruebas dobles para ABAP SQL, CDS y tablas de base de datos y pruebas de código heredado. Los seams de test se utilizan aquí para sustituir bloques del código productivo existente por un
00:09:33	código compatible con tests durante la ejecución de tests de unidad ABAP. Como puede ver en el panorama general de RAP, el lenguaje ABAP y el lenguaje de manipulación de
00:09:48	entidad se utilizan para la implementación del comportamiento de BO. Veamos ahora OData.
00:09:56	OData es un protocolo RESTful que permite un acceso simplificado a los datos de SAP. Se utiliza en una variedad de aplicaciones y tecnologías dentro de la cartera de productos y
00:10:09	tecnología de SAP, por ejemplo, en SAP S/4HANA, SAP Cloud Platform, SAPUI5 y SAP Fiori. OData está impulsado por Microsoft, SAP e IBM y se basa en estándares industriales como
00:10:27	HTTP, XML,
00:10:27	Atom y JSON. Además de la comunicación REST simple, OData define un contrato entre el servidor y el cliente
00:10:35	que se puede utilizar para consultar un servicio de forma similar a SQL. OData es ampliable y se puede ampliar para cumplir con los requisitos de las aplicaciones móviles y
00:10:50	basadas en navegador. Como puede ver en el panorama general de RAP, el protocolo OData se utiliza para la comunicación
00:11:01	externa para la vinculación de servicios y el consumo de servicios. SAP Fiori establece el estándar para la interfaz de usuario empresarial eliminando la complejidad
00:11:13	innecesaria. Pone a los usuarios en control de sus tareas empresariales dándoles solo lo que realmente
00:11:19	necesitan. Este objetivo central se refleja en cinco principios de diseño: basado en roles, adaptativo,
00:11:26	coherente, simple y agradable. Estos principios de diseño se pueden implementar con tecnologías de IU como SAP-UI5 para
00:11:34	aplicaciones basadas en navegador y SAP Fiori for Mobile para crear aplicaciones nativas en iOS y Android.
00:11:41	Los clientes y socios tienen diferentes opciones de desarrollo para crear sus propias aplicaciones. Desarrollo de código bajo basado en plantillas con elementos de SAP Fiori o
00:11:55	desarrollo de estilo
00:11:55	libre con SAPUI5 o aplicaciones móviles nativas para una flexibilidad total. Las herramientas Fiori admiten el desarrollo eficiente de las IU.
00:12:09	Se proporciona soporte especial para el desarrollo de aplicaciones de SAP Fiori proporcionando información adicional como ayuda para entradas y una vista previa de la aplicación de elementos
00:12:17	Fiori en las herramientas de desarrollo ABAP. Para el desarrollo de aplicaciones de elementos de SAP Fiori, es posible enriquecer el modelo de

- 00:12:24 datos con anotaciones de IU en el back end mediante ampliaciones de metadatos CDS. Sin embargo, el desarrollo típico de IU tiene lugar en las herramientas de SAP Fiori de SAP
- 00:12:38 Business Application Studio en SAP Cloud Platform. Las herramientas de desarrollo ABAP, en pocas palabras ADT, en la conocida plataforma de
- 00:12:49 desarrollo Eclipse constituye el entorno de desarrollo ABAP estándar para todas las tareas de desarrollo estándar on-premise, así como en SAP Cloud Platform.
- 00:12:58 Es el sucesor del Workbench ABAP. ADT ofrece un conjunto de herramientas de desarrollo moderno para aumentar la productividad
- 00:13:07 de los desarrolladores con muchas ventajas, como el resaltado de sintaxis, la finalización de códigos, las plantillas de código, las capacidades de búsqueda avanzada y las correcciones rápidas.
- 00:13:17 Las potentes herramientas de garantía y soporte de calidad también están integradas en ADT. Esto incluye una variedad de verificaciones de código con ABAP Test Cockpit y SAP Code
- 00:13:29 Vulnerability Analyzer, capacidades de prueba de unidad, así como funciones avanzadas de depuración, creación de perfiles y supervisión del tiempo de ejecución y análisis.
- 00:13:40 En consecuencia, ADT proporciona el mejor entorno de desarrollo integrado para todas las tareas de implementación de RAP.
- 00:13:55 Ahora estamos al final de esta unidad. Permítanme hacer una breve recapitulación.
- 00:13:59 En esta unidad, ha aprendido sobre las tecnologías vitales que hacen que el modelo de programación de aplicaciones RESTful ABAP y el papel que desempeñan en el panorama general.
- 00:14:10 Para más información, véase el apéndice de la presente presentación. Gracias por escuchar y verle en la siguiente unidad, Presentación de SAP Cloud Platform,
- 00:14:21 Entorno ABAP.

Week 1 Unit 4

- 00:00:05 Hola, y bienvenido al curso de openSAP Building Apps con ABAP RESTful Application Programming Model, semana uno, unidad cuatro, Presentación de SAP Cloud Platform, Entorno ABAP.
- 00:00:16 Me llamo Carine Tchoutouo Djomo y soy gestor de productos para la plataforma ABAP. En la unidad anterior, aprendió sobre las distintas tecnologías y herramientas implicadas en el modelo de
- 00:00:28 programación de aplicaciones RESTful ABAP. Como aprendió en la unidad uno, RAP está disponible en SAP S/4HANA y SAP Cloud Platform para el desarrollo de
- 00:00:38 aplicaciones ABAP. En esta unidad, le daré un resumen del entorno ABAP de SAP Cloud Platform que utilizaremos durante todo este
- 00:00:50 curso para los distintos ejercicios prácticos. ¿Por qué SAP ofrece ABAP como un entorno de desarrollo en SAP Cloud Platform?
- 00:01:01 En los últimos años, de hecho, décadas, una comunidad enorme ha formado alrededor de ABAP como lenguaje de programación.
- 00:01:09 Con millones de desarrolladores ABAP registrados en todo el mundo, ABAP ha evolucionado a un estándar de la industria para los productos de SAP utilizados por un gran número de clientes y partners en todo el
- 00:01:22 mundo. Dado que el software de SAP ofrece grandes posibilidades para adaptarlo a las necesidades
- 00:01:29 empresariales individuales, los desarrolladores ABAP han utilizado la plataforma para ampliar sus soluciones SAP mediante el desarrollo de un código personalizado ABAP.
- 00:01:40 Hay más de cien mil soluciones productivas basadas en ABAP con diferentes configuraciones que utilizan los clientes y los partners para diversos propósitos y para ejecutar procesos críticos para el negocio.
- 00:01:56 La plataforma ABAP ha demostrado ser un entorno que puede cumplir requisitos críticos y complejos. Se utiliza para crear aplicaciones listas para la empresa que amplían procesos y datos clave, utilizando
- 00:02:10 los conocimientos empresariales de los desarrolladores de ABAP. Combinar estos factores con la velocidad y la facilidad que proporciona ABAP como entorno de
- 00:02:20 desarrollo lleva a un costo competitivo de desarrollo y tiempo de llegada al mercado para las aplicaciones y extensiones personalizadas.
- 00:02:29 El mercado está experimentando una creciente adopción de soluciones de SAP basadas en la nube, como SAP S/4HANA Cloud.
- 00:02:37 Dado que estas soluciones ofrecen un alcance limitado en el área de extensibilidad, porque son gestionadas por SAP, el entorno ABAP ayuda a los clientes y socios en su transición a la nube permitiéndoles crear
- 00:02:54 extensiones de lado a lado basadas en ABAP en SAP Cloud Platform, consumiendo datos de las aplicaciones empresariales centrales a través de API.
- 00:03:06 SAP Cloud Platform, entorno ABAP y su ecosistema constan de varias herramientas y componentes de infraestructura que le permiten servir a una gran variedad de casos de uso.
- 00:03:17 Las principales categorías de la arquitectura de referencia son el conjunto de herramientas de desarrollador y administrador, el propio entorno ABAP, el ecosistema de SAP Cloud Platform y la conectividad
- 00:03:32 con sistemas y API externos. Empezando por la parte izquierda de la imagen, puede ver el conjunto de herramientas que admite
- 00:03:41 desarrolladores y administradores. El cockpit de SAP Cloud Platform proporciona al administrador de la nube acceso al entorno ABAP
- 00:03:50 mediante la IU de administración basada en web estándar para todos los servicios de SAP Cloud Platform.

- 00:03:58 A través del cockpit en la nube, se pueden realizar tareas como iniciar y borrar sistemas de entorno ABAP o configurar proveedores de identidades y opciones de seguridad.
- 00:04:09 Las herramientas de desarrollo ABAP para Eclipse, en pocas palabras ADT, proporcionan el conjunto de herramientas central para los desarrolladores de ABAP.
- 00:04:16 Ofrecen acceso directo a los sistemas de entorno ABAP y sirven como tiempo de diseño para todo el código ABAP que se crea.
- 00:04:26 El desarrollo de IU no se lleva a cabo dentro de ADT, sino mediante herramientas adicionales como las herramientas de SAP Fiori en SAP Business Application Studio.
- 00:04:38 La infraestructura Git permite a los desarrolladores importar y exportar códigos y gestionar transportes entre diferentes sistemas de entorno ABAP mediante el Sistema de modificaciones y transporte habilitado para
- 00:04:52 Git. En el centro de la imagen, puede ver el entorno ABAP que se ejecuta en SAP Cloud Platform.
- 00:05:00 Consiste en la plataforma ABAP y la base de datos SAP HANA debajo y utiliza solo las funciones más recientes y más importantes, como el modelo de programación de aplicaciones RESTful ABAP, así como un lenguaje ABAP
- 00:05:14 optimizado en la nube. Ofrece acceso a la oferta de microservicios de SAP Cloud Platform a través de su infraestructura de
- 00:05:24 conectividad. La misma conectividad se puede utilizar para acceder a servicios externos y API, lo que nos lleva al lado
- 00:05:35 derecho de la imagen. El entorno ABAP de SAP Cloud Platform le permite conectarse a sistemas externos y API de varias
- 00:05:46 maneras. Las soluciones SAP Cloud se pueden ampliar a través de las API OData de lanzamiento.
- 00:05:52 Estas API se documentaron en SAP API Hub. Se puede acceder a las API web siempre que sean compatibles con HTTPS o el protocolo OData.
- 00:06:06 Esto permite que el entorno ABAP también pueda lidiar con API externas procedentes de sistemas o servicios externos a SAP. Los sistemas locales como SAP S/4HANA y SAP Business Suite, o cualquier otro sistema SAP, se
- 00:06:22 pueden ampliar mediante HTTPS, OData o el protocolo RFC. El conector en la nube de SAP se utiliza aquí como herramienta estándar para establecer conexiones
- 00:06:34 seguras entre la infraestructura local y SAP Cloud Platform. Echemos un vistazo a los tres escenarios de ampliación principales admitidos con SAP Cloud Platform, entorno
- 00:06:50 ABAP. Con el primer escenario de ampliación, puede utilizar el entorno ABAP para ampliar SAP S/4HANA Cloud u otras
- 00:06:57 soluciones empresariales de SAP Cloud con ampliaciones en la nube personalizadas. Las soluciones SAP Cloud ofrecen opciones limitadas de ampliabilidad dentro de la aplicación para ampliar los
- 00:07:08 procesos y las aplicaciones estándar de SAP. Por ejemplo, SAP S/4HANA Cloud proporciona capacidades de ampliabilidad de usuario clave dentro de la
- 00:07:17 aplicación, pero no admite el desarrollo ABAP personalizado clásico. SAP Cloud Platform proporciona la base para desarrollar y ejecutar ampliaciones en la nube
- 00:07:29 personalizadas para estas soluciones y el entorno ABAP se utiliza aquí para proporcionar ampliaciones en la nube basadas en ABAP.
- 00:07:39 Con el segundo escenario de ampliación, puede desarrollar y ejecutar aplicaciones ABAP completas e innovadoras en la nube.
- 00:07:47 Se beneficiará de la plataforma ABAP más reciente y de las tecnologías de base de datos de SAP HANA, independientemente de su entorno de sistema on-premise existente.
- 00:07:58 Desarrolle aplicaciones de SAP Fiori de última generación con el RAP a prueba de futuro. Utilice fácilmente servicios de SAP Cloud Platform como IoT y machine learning en sus apps.

- 00:08:13 Y delegue la operación de la Plataforma ABAP como servicio a SAP. Y, por último, con el tercer escenario de ampliación, puede desvincular sus implementaciones ABAP
- 00:08:29 personalizadas de sus sistemas empresariales centrales locales y tener un uso similar al de hubs del entorno ABAP.
- 00:08:37 Puede poner sus aplicaciones en la nube a disposición de un público externo que no tenga acceso a sus sistemas empresariales centrales.
- 00:08:45 Este es, por ejemplo, el caso típico de las aplicaciones de consumidor. Puede utilizar el entorno ABAP como un hub de integración integrando múltiples sistemas en la nube y
- 00:08:58 on-premise con servicios en la nube de SAP y no SAP en sus extensiones en la nube. Otro caso de uso aquí es la integración de datos.
- 00:09:07 Puede recopilar datos de varias fuentes en sus extensiones en la nube para su procesamiento y análisis posteriores.
- 00:09:18 Las extensiones en la nube son extensiones separadas que solo utilizan API remotas y locales bien definidas.
- 00:09:25 Gracias a esto, el sistema empresarial central se mantiene limpio y se reduce el riesgo y el trabajo implicado durante las actualizaciones.
- 00:09:39 Ahora estamos al final de esta unidad. Vamos a resumirlo con una breve recapitulación.
- 00:09:46 En esta unidad, ha aprendido sobre la oferta de plataforma como servicio ABAP de SAP, el entorno ABAP de SAP Cloud Platform.
- 00:09:57 Ha aprendido cuáles son las partes vitales, las ventajas y los principales escenarios de ampliación del entorno ABAP.
- 00:10:06 Para más información sobre esta unidad, véase el apéndice de la presente presentación. Gracias por escuchar y verle en la siguiente unidad, donde prepararemos el entorno de desarrollo ABAP.

Week 1 Unit 5

- 00:00:05 Hola y bienvenido al curso de openSAP "Building Apps with the ABAP RESTful Application Programming Model", semana uno, unidad cinco, "Preparación de su entorno de desarrollo ABAP".
- 00:00:17 Me llamo Carine Tchoutouo Djomo y soy gestor de productos para la plataforma ABAP. En la unidad anterior, tuvimos una introducción a SAP Cloud Platform, entorno ABAP, la oferta de plataforma como servicio de
- 00:00:30 SAP para el desarrollo de aplicaciones ABAP en la nube. En esta unidad, configuraremos nuestro entorno de desarrollo ABAP.
- 00:00:42 Utilizaremos la oferta de prueba del entorno ABAP para los diversos ejercicios prácticos de este curso de openSAP, pero tenga en cuenta que RAP está disponible en SAP Cloud Platform y en SAP S/4HANA.
- 00:00:57 Para este curso, necesitará una cuenta de entorno ABAP de SAP Cloud Platform. Si aún no tiene una, puede registrarse para nuestra oferta de prueba gratuita, que forma parte de la prueba general de SAP Cloud
- 00:01:12 Platform Cloud Foundry. Una cuenta de prueba caduca después de 30 días, pero se puede ampliar a un máximo de 90 días.
- 00:01:22 Un sistema de prueba tiene una vida útil de cuatro meses y puede incorporarse en el primer mes. Por lo tanto, todo el mundo puede acceder a un sistema de acceso de prueba al menos durante tres meses.
- 00:01:34 Después de este período, se borran los sistemas. Por lo tanto, no se recomienda aplicar ningún código que sea urgentemente necesario después del período de prueba.
- 00:01:44 Se aplican varias reglas a la oferta de prueba. Se destina únicamente a fines educativos.
- 00:01:52 Los sistemas de prueba del entorno ABAP son instancias compartidas y no hay separación de contenido entre los usuarios. Significa que varios usuarios de prueba pueden acceder al mismo sistema y pueden ver el código del otro.
- 00:02:09 SAP ofrece un sitio de actualización central para la instalación de las diferentes herramientas de desarrollo de SAP basadas en Eclipse, incluidas las herramientas de desarrollo ABAP, en un breve ADT.
- 00:02:21 Allí encontrará todas las instrucciones necesarias sobre la instalación local de la plataforma Eclipse y el complemento ADT. Así que pongámonos manos a la obra.
- 00:02:35 Primero incorporaremos a la prueba de entorno ABAP. Como ya se ha mencionado anteriormente, necesita una cuenta de prueba de SAP Cloud Platform Cloud Foundry para hacerlo.
- 00:02:45 Si aún no tiene una cuenta en el sitio web de SAP, primero tendrá que registrarse en SAP.com y activar su cuenta gratuita. Vamos a navegar a la página de prueba.
- 00:03:02 En la página de prueba de SAP Cloud Platform, proporcione su correo electrónico y contraseña registrados y pulse Iniciar sesión.
- 00:03:19 Ahora estamos en la pantalla de bienvenida. Haga clic en "Introducir su cuenta de prueba" para navegar a su cuenta global.
- 00:03:26 Si esta es la primera vez que accede a su cuenta global, tendrá que seleccionar una región geográfica de su elección y hacer clic en Crear cuenta para iniciar el proceso de aprovisionamiento.
- 00:03:38 Su cuenta global contendrá la subcuenta de prueba creada y el espacio. Navegue a la subcuenta de prueba haciendo clic en la hora llamada Prueba.
- 00:03:48 El nombre puede variar si ha creado la subcuenta de prueba manualmente. En Cloud Foundry, cada aplicación y servicio vive en un espacio, por lo que navega hasta el espacio de desarrollo en el que sus
- 00:04:19 aplicaciones y servicios viven haciendo clic en él. En el panel de navegación, en la parte izquierda, haga clic en Service Marketplace para navegar al resumen de servicios donde

00:04:33 puede seleccionar Prueba ABAP como servicio. Haga clic en el mosaico Prueba ABAP.

00:04:42 Ahora estamos en la página de resumen del servicio de prueba. Para tener acceso de prueba a un entorno ABAP compartido, seleccione Instancias en la parte izquierda y haga clic en Nueva

00:04:56 instancia. Proporcione la información necesaria en el asistente.

00:05:00 Este es un plan de servicios compartidos, por lo tanto, mantenga la entrada predeterminada y haga clic en Siguiente. Introduzca su dirección de correo electrónico de registro entre comillas y haga clic en Siguiente.

00:05:24 No es necesario vincular una aplicación para la instancia de servicio, por lo tanto, conserve la entrada predeterminada y haga clic en Siguiente.

00:05:32 Proporcione un nombre a su instancia, normalmente su nombre de usuario, y haga clic en Finalizar para finalizar la creación de la instancia.

00:05:48 Su instancia de prueba ABAP se ha creado y aparece en la página de resumen de instancias. Como paso siguiente, necesitamos crear una clave de servicio para configurar manualmente la conexión a la instancia de prueba del

00:06:00 entorno ABAP en las herramientas de desarrollo ABAP más adelante. Para ello, haga clic en su instancia y seleccione Claves de servicio en la parte izquierda del panel de navegación.

00:06:11 Ahora haga clic en Crear clave de servicio para crearla. Introduzca un nombre y haga clic en Guardar.

00:06:22 Se ha creado su clave de servicio y ahora aparece en la página. Tiene un formato JSON.

00:06:31 Puede copiar la clave de servicio y descargarla para su uso posterior. Ahora estamos incorporados a la prueba de entorno ABAP de SAP Cloud Platform.

00:06:43 Ahora instalemos nuestras herramientas de desarrollo ABAP locales en Eclipse. SAP proporciona un sitio de actualización central para todas las herramientas de desarrollo ABAP basadas en Eclipse.

00:06:58 Seleccione la pestaña ABAP y allí encontrará los requisitos previos y las instrucciones relacionadas con la instalación. En la sección Procedimiento , encontrará descripciones paso a paso de lo que debe hacer. Primero, debe instalar la versión de Eclipse

00:07:19 indicada. Ya he instalado Eclipse 2020-06 compatible actualmente en mi equipo y seguimos adelante con los siguientes pasos.

00:07:29 Por lo tanto, lo que haré aquí ahora, copiaré la URL indicada. Así que pasemos a mi IDE.

00:07:40 Aquí está mi instalación local de Eclipse. En cualquier momento puede ver la versión de los componentes de software instalados en ella.

00:07:51 Como puede ver, podemos ver la versión, pero también los componentes de software de pedido o los complementos instalados. Cierre eso.

00:07:59 Para iniciar la instalación de ADT, vamos a la barra de menús y seleccionamos Ayuda, Instalar nuevo software. Y aquí proporcionaremos una nueva entrada.

00:08:12 Llamémoslo Herramientas de desarrollo ABAP, y proporcionemos la URL, cópiela del sitio de actualización central. Solo añadiré.

00:08:26 Y muy pronto veremos todo el software que está disponible para descargar. Puede tardar unos segundos.

00:08:37 Aquí aparece que tengo las herramientas de desarrollo ABAP y otras herramientas como las herramientas de SAP HANA o las herramientas de modelado.

00:08:45 A continuación, haga clic aquí y también puedo abrir el árbol. A continuación, haga clic en Siguiente.

00:08:59 De nuevo, obtengo un resumen detallado de las diferentes herramientas que se instalarán y solo hago clic en Siguiente. Y aquí, tengo que aceptar las condiciones del contrato de licencia y hacer clic en Finalizar para iniciar la instalación del

00:09:19 complemento ADT. Aquí, en la barra de progreso, puedo ver que el software se está instalando.

- 00:09:26 Esperemos dos segundos. Ya hemos terminado, y ahora tenemos que reiniciar ADT para aplicar la actualización del software.
- 00:09:38 Esto puede tardar unos segundos antes del reinicio, antes de reiniciar el IDE. Ahora se reinicia el IDE y ya puedo ver información sobre mi release ADT, y el siguiente paso que debo hacer aquí es abrir la
- 00:10:09 perspectiva de las herramientas de desarrollo ABAP. Para ello, vaya a la barra de menús y vaya a la entrada de menú Windows, Perspective, Open Perspective y luego Other.
- 00:10:27 Aquí puede seleccionar ABAP Para cambiar a la perspectiva de desarrollo ABAP. Aquí se proporciona un acceso rápido a esta función en la barra de perspectiva donde puedo hacer clic en Abrir, y aquí puedo ver las
- 00:10:43 diferentes perspectivas que ya se han abierto con el icono. Y también puede cambiar fácilmente entre ambos directamente desde aquí.
- 00:10:55 El layout ADT estándar incluye el Explorador de proyectos, que proporciona una vista jerárquica de los proyectos ABAP. Tenemos el área de editor en el centro de nuestro IDE.
- 00:11:11 Tenemos la vista Esquema y también la vista Explorador de funciones y varias otras vistas útiles que se apilan debajo del área del editor.
- 00:11:26 Veremos más de cerca el ADT en la siguiente unidad. Antes de poder empezar a utilizar ADT, necesitamos conectarlo a nuestra instancia de prueba de entorno ABAP creando un proyecto
- 00:11:43 ABAP en la nube. En el contexto de ADT, un proyecto representa la conexión del sistema a nuestro back end ABAP.
- 00:11:51 Por lo tanto, un proyecto de nube ABAP se utiliza para conectar ADT a un sistema ABAP en SAP Cloud Platform, mientras que los proyectos ABAP, un proyecto ABAP se utiliza para la conexión a un sistema local.
- 00:12:05 Dado que el Explorador de proyectos está vacío, podemos empezar directamente desde aquí. Pero también es posible aquí.
- 00:12:13 Acceso rápido para crear el proyecto, o desde "Archivo" y luego "Nuevo". Por lo tanto, vamos a crear nuestro proyecto de nube ABAP.
- 00:12:23 Para ver el asistente que aparece, seleccionaremos Clave de servicio. Si recuerda durante la incorporación, hemos creado nuestro kit de servicio y aquí podemos introducirlo directamente, si lo copiamos
- 00:12:41 de la cuenta de prueba de SAP Cloud Platform o importarlo si lo descargamos. Ahora se importa y podemos continuar haciendo clic en Siguiente.
- 00:12:56 Y, si se solicita, proporcione su correo electrónico y contraseña de prueba registrados para autenticarse. Ahora hemos iniciado sesión correctamente en la prueba de back end ABAP.
- 00:13:14 Simplemente haga clic en Finalizar. La conexión con el sistema de prueba ABAP ahora aparece en el Explorador de proyectos.
- 00:13:27 Desde aquí, podemos explorar los objetos de desarrollo ABAP en el repositorio del sistema ABAP y abrir los objetos que deseamos en el editor correspondiente.
- 00:13:38 Ahora estamos todos preparados. Eso fue para la demostración del sistema.
- 00:13:43 Ahora estamos al final de esta unidad. Permítanme hacer una breve recapitulación.
- 00:13:47 En esta unidad, ha aprendido a configurar su entorno de desarrollo ABAP, cómo obtener acceso a la cuenta de prueba del entorno ABAP de SAP Cloud Platform, cómo instalar las herramientas de desarrollo ABAP en Eclipse y cómo conectarse al sistema de prueba.
- 00:14:06 Para más información sobre esta unidad, véase el apéndice de la presente presentación. Gracias por escucharle y verle en la siguiente unidad, donde crearemos nuestra primera aplicación de consola en la nube ABAP.

Week 1 Unit 6

- 00:00:05 Hola, y bienvenido al curso de openSAP "Building Apps with the ABAP RESTful Application Programming Model", semana uno, unidad seis, para crear su primera aplicación de consola ABAP en la nube.
- 00:00:16 Me llamo Carine Tchoutouo Djomo y soy gestor de productos para la plataforma ABAP. En la unidad anterior, preparamos nuestro entorno de desarrollo ABAP incorporando la versión de prueba del entorno ABAP de SAP Cloud Platform, instalando nuestro IDE local, las
- 00:00:33 herramientas de desarrollo ABAP y conectándolo al sistema de prueba. En esta unidad, vamos a seguir adelante y escribir nuestra primera aplicación de consola en la nube ABAP.
- 00:00:48 Crearemos un ABAP Hola, ¡Mundo! aplicación de consola, y también veremos algunas de las buenas funciones de ADT que ayudan a acelerar el desarrollo. Así que pongámonos manos a la obra.
- 00:01:03 Antes de implementar la aplicación de consola, analicemos un poco las herramientas de desarrollo ABAP y algunas de sus funciones, que ayudan a acelerar el desarrollo de ABAP en Eclipse.
- 00:01:14 Vamos a cerrar algunas de las ventanas y echemos un vistazo a las vistas comprendidas en el layout por defecto de la perspectiva ABAP. Primero, la vista Explorador de proyectos , que proporciona una vista jerárquica de sus proyectos ABAP.
- 00:01:28 Desde aquí, puede explorar el objeto de desarrollo ABAP en este sistema ABAP y abrir sus objetos de desarrollo en el editor correspondiente. Puede organizar el objeto de desarrollo de acuerdo con su trabajo actual utilizando árboles de repository ABAP.
- 00:01:47 ADT ofrece los árboles predefinidos Paquetes favoritos, Objetos favoritos y Objetos liberados por defecto. En los paquetes favoritos, actualmente podemos ver ZLOCAL, el paquete generado para todos los objetos de desarrollo local en el entorno ABAP de prueba.
- 00:02:06 Puede añadir objetos de desarrollo relevantes para su trabajo a Paquetes favoritos y Objetos favoritos utilizando el menú contextual adecuado en cualquier momento para acceder fácilmente a ellos o eliminarlos.
- 00:02:20 También puede definir árboles de repository ABAP nuevos utilizando varios criterios de filtro. Asegúrese de que la vista está vinculada con el área Editor mediante el botón Enlazar con editor.
- 00:02:35 Al hacerlo, cada vez que modifique un objeto ABAP en el editor, este objeto se seleccionará en la exploración de proyecto expandida. En el centro, tenemos el área Editor.
- 00:02:53 Aquí es donde se abren los editores para los objetos de desarrollo ABAP y se proporcionan editores basados en Eclipse para todas las tareas de implementación estándar. En el Explorador de proyectos, tenemos la vista Esquema , que muestra la estructura interna del objeto de desarrollo ABAP seleccionado abierto en el área Editor.
- 00:03:16 En la parte derecha, tenemos la vista Explorador de funciones , que proporciona un tutorial interactivo para ayudar a los desarrolladores a familiarizarse con las funciones básicas de ADT.
- 00:03:27 Otras vistas útiles se retienen debajo del área Editor en un container de fichas. Dos de las vistas más utilizadas aquí son la vista Problemas , que muestra errores de sintaxis, advertencias y otra información asociada con el objeto de desarrollo ABAP abierto
- 00:03:48 en el área Editor , y una vista Propiedades , que muestra las propiedades básicas del objeto de desarrollo ABAP seleccionado en el Explorador de proyectos. Otra cosa que puede hacer es personalizar el diseño de las perspectivas en Eclipse como desee.
- 00:04:10 No hay casi ningún límite en la modificación de la disposición. Por ejemplo, puede reorganizar el diseño, cerrar vistas, abrir nuevas, modificar el tamaño de la vista y mover las vistas.

- 00:04:23 Por ejemplo, cerremos una vista y añadamos una nueva. Para ello, para agregar uno nuevo, vamos a la barra de menús ADT y abrimos el menú Windows y seleccionamos Mostrar vistas.
- 00:04:47 Y allí puede elegir entre los valores visualizados en el menú o pulsar Otro y buscar la vista deseada. Añadamos la vista Explorador de relaciones , que le ayuda a comprender cómo los objetos de desarrollo están relacionados entre sí.
- 00:05:07 En el contexto RAP, el Explorador de relaciones muestra la estructura y el comportamiento de los business objects RAP, independientemente de la ubicación técnica. No olvide vincular la nueva vista al área del editor, para tener siempre una sincronización automática entre ambos, utilizando el botón Enlazar con el editor.
- 00:05:28 También puede reorganizar la posición del editor. En cualquier momento, puede restablecer una perspectiva a su diseño predeterminado navegando a la entrada de menú en Windows, Perspectiva, Restablecer perspectiva en el menú.
- 00:05:52 O simplemente haciendo clic con el botón derecho en el icono de perspectiva relevante en la barra de perspectiva en la esquina superior derecha y seleccionando Reinicializar. Otro consejo útil para aumentar la productividad del desarrollador es el uso de accesos directos ABAP.
- 00:06:12 Presione el acceso directo del teclado CTRL + Shift + L en ADT para obtener un resumen de todos los accesos directos del teclado disponibles que se muestran en una pequeña ventana. Además de los utilizados para guardar y activar objetos de desarrollo, algunos de los accesos directos ADT utilizados con más frecuencia son CTRL + Espacio para desencadenar la
- 00:06:35 finalización del código, CTRL + Mayús + A para abrir un nuevo objeto de desarrollo, o F2 para mostrar la información del elemento. Además, con CTRL + 1, puede desencadenar correcciones rápidas con una vista de ayuda rápida para abrir y utilizar las correcciones rápidas disponibles.
- 00:06:56 Puede encontrar más información sobre ADT en la documentación ABAP, a la que se puede acceder a través del menú, Ayuda, Contenido de ayuda Allí también encontrará las notas de release de ADT, pero también varias guías de desarrollo, por ejemplo, para ADT, el modelo de programación de aplicación RESTful ABAP o para los servicios de datos de llamada ABAP.
- 00:07:17 Ahora vamos a crear la aplicación de consola. Primero, crearemos un nuevo paquete ABAP para agrupar los diferentes objetos de desarrollo que vamos a crear durante este curso.
- 00:07:31 Para ello, vaya al proyecto ABAP y abra su menú contextual haciendo clic con el botón derecho en él y creando el paquete ABAP. Tenemos que proporcionar un nombre y una descripción, podemos mantener las otras entradas predeterminadas y pulsar Siguiente para continuar.
- 00:07:53 Ahora tenemos que proporcionar una solicitud de transporte, si tiene una existente, puede seleccionarla o simplemente crear una nueva, proporcionar una descripción de solicitud y seleccionar Finalizar.
- 00:08:14 Ahora se crea el nuevo paquete y, como paso siguiente, lo añadiré a Mis paquetes favoritos. El paquete ahora se añade a la lista.
- 00:08:32 Así que ahora podemos seguir adelante y crear nuestro Hola, ¡Mundo! aplicación de consola. Para ello, haga clic con el botón derecho en el paquete y, en el asistente de creación, el nombre del proyecto y el nombre del paquete se incluirán automáticamente.
- 00:08:49 Introduzca un nombre y una descripción para la clase ABAP y continúe con Siguiente. Y aquí conservaré la selección de la orden de transporte y pulsaré Finalizar.
- 00:09:10 La clase ABAP se crea y se abre en el área Editor de clase basada en fuente, lista para implementar. ¡Hola, mundo! La clase debe implementar una interfaz ABAP especial para escribir la salida en la consola ABAP.
- 00:09:24 Añadamos la interfaz necesaria en la sección pública de la definición de clase. También puede utilizar la compleción de código pulsando CTRL + Espacio.

- 00:09:41 Por lo tanto, la interfaz se inserta y tenemos una advertencia que aparece, por lo que podemos pasar el ratón sobre el icono para leer el texto o ir a la vista Problemas. Y aquí vemos que el motivo de esta advertencia es la implementación del método que falta actualmente.
- 00:09:57 Y al utilizar CTRL + 1, tenemos la vista de ayuda rápida y aquí podemos buscar propuestas y añadir automáticamente la implementación del método. Ahora podemos seguir adelante e implementar nuestro Hola, ¡Mundo! clasificación.
- 00:10:20 Estamos utilizando un modelo de cadena con texto literal. Puede pulsar Mayús + 1 para dar formato al código fuente.
- 00:10:34 Puede acceder a la configuración del formateador de código fuente en el menú Windows, Preferencias, y puede buscar formateador. Haga clic en la Edición de código fuente y, a continuación, haga clic en el enlace de formateador ABAP; esta es una opción específica de proyecto, así que seleccione el proyecto y
- 00:10:57 puede configurarlo como desee. Y pulse Aplicar y Cerrar.
- 00:11:03 Otro punto es la configuración de finalización de código. Esto también es muy interesante.
- 00:11:11 Aquí puede decidir, por ejemplo, para el asistente de contenido, si la finalización se inserta o si la finalización sobrescribe el texto existente. He elegido sobrescribir la opción.
- 00:11:25 Ahora puedo presentar mi candidatura y luego ir más allá. Ahora puedo guardar, activar mi clase y ejecutarla como una consola de aplicación.
- 00:11:37 ¡Estupendo, podemos ver la Hola, Mundo! salida en la consola ABAP, así que estamos al final. Existen diferentes formas de ejecutar una aplicación como una consola ABAP.
- 00:11:54 Ahora estamos casi al final. Ampliemos la salida actual con nuestro nombre de usuario o alias de usuario.
- 00:12:02 Para ello, necesitamos acceder a las variables del sistema. En el entorno ABAP, varias variables de sistema ya no están disponibles debido al entorno de nube o a los marcos no admitidos.
- 00:12:14 Para acceder, liberar, se proporcionan las variables del sistema como el nombre de usuario, el idioma y el tiempo, se proporcionan API que se deben utilizar. Por lo tanto, ampliemos la plantilla de cadena actual con una expresión incrustada y la clase de ayuda `abap_context_info`.
- 00:12:42 Podemos ver los diferentes métodos Get disponibles en esta clase, seleccionar `get_user_alias` y ahora podemos guardar, activar y ejecutar de nuevo la clase ABAP mejorada. De este modo, ahora podemos ver la salida mejorada en la consola ABAP.
- 00:13:07 Puede borrar la consola ABAP con su menú contextual. Eso es todo.
- 00:13:20 Eso fue para la demostración. Ahora estamos al final de esta unidad.
- 00:13:25 Tratémoslo con una breve recapitulación. En esta unidad, ha aprendido a crear un ABAP Hola, Mundo. aplicación de consola, cómo utilizar las herramientas de desarrollo ABAP, cómo crear paquetes y clases ABAP y cómo
- 00:13:41 activar y ejecutar un programa en ABAP. Para más información sobre esta unidad, véase el apéndice de la presente presentación.
- 00:13:50 Hemos llegado al final de la semana uno del curso de openSAP Creación de aplicaciones con el modelo de programación de aplicaciones RESTful ABAP. Gracias por escuchar y verle la próxima semana donde desarrollaremos una aplicación de informe de lista de SAP Fiori de solo lectura.

www.sap.com/contactsap

© 2020 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company.

The information contained herein may be changed without prior notice. Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

In particular SAP SE or its affiliated companies have no obligation to pursue any course of business outlined in this document or any related presentation, or to develop or release any functionality mentioned therein. This document, or any related presentation, and SAP SE's or its affiliated companies' strategy and possible future developments, products, and/or platform directions and functionality are all subject to change and may be changed by SAP SE or its affiliated companies at any time for any reason without notice. The information in this document is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, and they should not be relied upon in making purchasing decisions.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies. See www.sap.com/copyright for additional trademark information and notices.